
(19)

KOREAN INTELLECTUAL PROPERTY OFFICE

KOREAN PATENT ABSTRACTS

(11)Publication

1020020085674 A

number:

(43)Publication date: 16.11.2002

(21)Application
number:

1020010025306

(22)Application date: 09.05.2001

(71)Applicant:

• SAMSUNG
ELECTRONICS
CO., LTD.

(72)Inventor:

• CHOI, HO GYU
• JANG, JAE
SEONG
• KIM, JAE YEOL
• KIM, YUN SEON
• KWON, HWAN
JUN

(51)Int. Cl

H03M 13/00

**(54) APPARATUS FOR ENCODING AND DECODING IN CODE DIVISION
MULTIPLE ACCESS MOBILE COMMUNICATION SYSTEM AND METHOD FOR
THE SAME**

(57) Abstract:

PURPOSE: An apparatus for encoding and decoding in a code division multiple access(CDMA) mobile communication system and a method for the same are provided to implement a data rate indicator encoding method and apparatus with an optimum code performance efficiency and a minimum complexity by using an apparatus and method for punching an expanded first order Reed Muller code, thereby obtaining an effect corresponding to the use of an optimum code language.

CONSTITUTION: An apparatus for encoding and decoding in a CDMA mobile communication system includes Walsh code generator(500), a mask generator(505), a controller(510), a plurality of multipliers(520,522,524,526,528,530,532), an exclusive adder(540), a repeater(550), a punch(560) and a memory(570). The apparatus outputs a coded symbol column consisting of 32 number of coded symbols as a coded symbol column consisting of 24 number of coded symbols, wherein the input information bits consisting of 6 bits are inputted to the multipliers(520,522,524,526,528,530,532). The apparatus obtains the 32 number of coded symbols by coding each of the input information bits consisting of 6 bits into 5 number of Walsh codes having a length of 32 and one mask by the exclusive adder(540). The punch(560) punches 0,4,8,12,16,20,24 and 28th symbols among the 32 number of coded symbols from the exclusive adder(540) to output the coded symbol column consisting of 24 coded symbols.

(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(51) 。 Int. Cl. 7
H03M 13/00

(11) 공개번호 특2002- 0085674
(43) 공개일자 2002년11월16일

(21) 출원번호 10- 2001- 0025306
(22) 출원일자 2001년05월09일

(71) 출원인 삼성전자 주식회사
경기 수원시 팔달구 매탄3동 416번지

(72) 발명자 김윤선
서울특별시강남구대치3동63우성아파트3- 1007
장재성
경기도과천시중앙동주공아파트1102동203호
최호규
경기도성남시분당구구미동무지개마을1204동303호
김재열
경기도군포시산본2동산본9단지백두아파트960동1401호
권환준
서울특별시강동구둔촌2동미도맨션1동203호

(74) 대리인 이건주

심사청구 : 없음

(54) 부호분할다중접속 이동통신시스템에서 부호화 및 복호화장치 및 방법

요약

본 발명은 부호분할다중접속 이동통신시스템에서 이동통신 방식에서 사용하는 채널중에 역방향 데이터를 지시채널(Reverse Rate Indicator Channel, R-RICH)을 전송하는 장치 및 방법에 관한 것으로, 최적 (24,1), (24,2), (24,3), (24,4), (24,5), (24,6) 및 (24,7) 부호화를 위한 최적의 부호어를 생성하고, (24,1), (24,2), (24,3), (24,4), (24,5), (24,6) 및 (24,7) 부호화를 겸용하여 지원할 수 있는 부호화 및 복호화 장치 및 방법을 구현하였다.

대표도
도 5

색인어
부호분할다중접속 이동통신시스템, 데이터를 지시채널, 천공, 부호화

명세서

도면의 간단한 설명

도 1은 통상적인 부호분할다중접속 이동통신시스템에서 역방향 데이터를 지시채널의 송신기 구조를 도시한 도면.

도 2는 본 발명의 실시 예에 따른 부호분할다중접속 이동통신시스템에서 최적 (24,4) 부호화기의 구성을 도시하고 있는 도면.

도 3은 본 발명의 실시 예에 따른 부호어 구성의 일 예를 보여주고 있는 도면.

도 4는 본 발명의 실시 예에 따른 부호분할다중접속 이동통신시스템에서 최적 (24,7) 부호화기의 구성을 도시하고 있는 도면.

도 5는 본 발명의 실시 예에 따른 부호분할다중접속 이동통신시스템에서 최적 (24,1) 내지 (24,7) 부호화기를 동시에 지원하는 부호화기의 구성을 도시하고 있는 도면.

도 6은 본 발명의 실시 예에 따른 부호분할다중접속 이동통신시스템에서 복호화기의 구성을 보여주고 있는 도면.

도 7은 본 발명의 실시 예에 따른 생성행렬을 이용한 (24,4) 부호에 대한 부호기의 구성을 보여주고 있는 도면.

도 8은 본 발명의 실시 예에 따른 생성행렬을 이용한 (24,7)에 대한 부호기의 구성을 보여주고 있는 도면.

도 9는 본 발명의 실시 예에 따른 부호분할다중접속 이동통신시스템에서 최적 (24,1) 내지 (24,7) 부호화기를 동시에 지원하는 부호화기의 다른 구성을 도시하고 있는 도면.

도 10은 본 발명의 실시 예에 따른 데이터 지시자에 할당되는 비트 수를 최소화하기 위한 역방향 데이터를 지시채널의 송신기 구조를 도시하고 있는 도면.

도 11은 본 발명의 실시 예에 따른 송신기에 구비된 (24,1) 부호화를 위한 구성을 도시한 도면.

도 12는 도 11에서의 생성행렬에 따른 (24,1) 부호에 대한 부호기를 도시한 도면.

도 13은 본 발명의 실시 예에 따른 송신기에 구비된 (24,2) 부호화를 위한 구성을 도시한 도면.

도 14는 도 13에서의 생성행렬에 따른 (24,2) 부호에 대한 부호기를 도시한 도면.

도 15는 본 발명의 실시 예에 따른 송신기에 구비된 (24,3) 부호화를 위한 구성을 도시한 도면.

도 16은 도 15에서의 생성행렬에 따른 (24,3) 부호에 대한 부호기를 도시한 도면.

도 17은 본 발명의 실시 예에 따른 송신기에 구비된 (24,5) 부호화를 위한 구성을 도시한 도면.

도 18은 도 17에서의 생성행렬에 따른 (24,5) 부호에 대한 부호기를 도시한 도면.

도 19는 본 발명의 실시 예에 따른 확장된 직교부호의 구조를 도시한 도면.

도 20은 본 발명의 실시 예에 따른 송신기에 구비된 (24,6) 부호화를 위한 구성을 도시한 도면.

도 21는 도 20에서의 생성행렬에 따른 (24,6) 부호에 대한 부호기를 도시한 도면.

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 부호분할다중접속 이동통신시스템에서의 부호화 및 복호화 장치 및 방법에 관한 것으로, 동기 이동통신 방식에서 사용하는 채널중에 역방향 데이터를 지시채널(Reverse Rate Indicator Channel, R- RICH)을 전송하는 장치 및 방법에 관한 것이다.

통상적으로 역방향 패킷 채널(Reverse Supplemental Channel, R- SCH)에서는 기본적으로 가변율 전송 방식을 지원하고 있다. 여기서, 가변율 전송 방식이란 단말이 전송할 데이터율을 임의로 변경할 수 있음을 의미한다. 일반적으로 데이터율이 달라지면 프레임 구성에 사용되는 오류 정정 부호의 부호율, 심볼 반복 횟수, 대역 확산을 위한 월시 부호 등의 길이 및 종류 등이 달라지게 된다. 따라서, 단말은 현재 전송하고 있는 역방향 패킷 채널의 데이터율을 기지국에 알려주어야만 기지국 수신기가 역방향 패킷채널을 제대로 수신할 수 있다. 이러한 용도를 위해 정의된 채널을 역방향 데이터율 지시채널(Reverse Rate Indicator Channel, R- RICH)이라 한다.

상기 역방향 패킷 채널을 통해 단말이 전송할 수 있는 데이터율의 가지 수는 단말이 동시에 사용할 수 있는 역방향 패킷 채널의 개수에 따라 달라지게 된다. 상기 역방향 패킷 채널의 개수는 호 설정시 역방향으로 전송할 데이터의 양 등을 고려하여 기지국에 의해 결정되어 단말에게 통보된다. 따라서, 역방향 데이터를 지시채널로 전송되는 정보 비트의 수는 역방향 패킷 채널의 개수에 따라 달라지게 된다. 즉, 역방향 패킷채널의 개수가 한 개일 때는 4비트를 이용하여 역방향 데이터율을 알려주게 되어 있으며, 역방향 패킷채널의 개수가 두 개일 때는 7비트를 이용하여 역방향 데이터율을 알려주도록 되어 있다. 한편, 단말이 동시에 사용할 수 있는 역방향 패킷 채널의 개수는 기지국에 의해 별도의 지시가 있기 전까지는 변할 수 없으므로 역방향 데이터율 지시채널로는 항상 4비트 정보가 전송되거나 항상 7비트 정보가 전송되게 된다. 즉, 상기 4비트 정보와 7비트 정보가 동시에 전송되는 경우는 없다. 이미 공지된 종래 기술에서는 상기 역방향 데이터율 지시채널에서 사용할 오류정정 부호로 (24,4) 또는 (24,7) 부호를 정의하고 있다.

상기 역방향 데이터율 지시 채널의 문제점은 역방향 데이터율을 전송하기 위하여 전송되는 비트의 개수가 역방향 패킷 채널의 개수에만 따라 달라진다는 것이다. 즉, 상기 역방향 패킷 채널을 통해 단말이 전송할 수 있는 데이터율의 가지 수가 전송되는 비트의 개수를 결정하는데 고려되지 않는다. 역방향 패킷 채널을 통해 단말이 전송할 수 있는 데이터율의 가지수와 무관하게 전송되는 비트의 개수를 결정할 경우 실제로 필요한 것보다 많은 비트수를 전송하는 경우가 발생할 수 있다. 예를 들어 역방향 패킷 채널이 한 개이고 역방향 패킷 채널을 통해 단말이 전송할 수 있는 데이터율의 가지 수가 네 가지일 경우 전송하는 필요한 최소의 비트 수는 2이다. 하지만 종래 기술에서는 역방향 패킷 채널이 한 개일 경우 데이터율을 전송하는데 필요한 최소의 비트수 보다 2 비트 많은 4 비트로 전송한다.

상기와 같이 데이터율 지시 채널로 전송할 비트의 개수를 역방향 패킷 채널의 개수에 따라 결정하는 종래의 방식은 필요 이상의 비트수를 데이터율 지시 채널에 전송하는 문제점이 있다. 데이터율 지시 채널에 필요 이상의 비트수를 전송하는 것은 부호화기의 부호화율을 높게 하여 최적의 부호화 방법을 적용하지 못하게 할 수 있다.

도 1은 상기 역방향 데이터율 지시채널의 송신기 구조를 나타내는 도면이다.

상기 도 1을 참조하여 설명하면, 먼저 4비트 또는 7비트의 데이터율 지시자는 부호기 100에 입력되어진다. 그러면, 상기 부호기 100은 상기 데이터율 지시자를 부호화하여 24개의 부호화 심볼을 출력하게 된다. 그러면, 상기 출력되어진 부호화 심볼들은 심볼반복기 110에 입력되고, 상기 심볼반복기 110은 상기 입력되어진 24개의 부호화 심볼들을 16번 만큼 반복하여 출력한다. 그러면, 상기 반복되어진 부호화 심볼들은 신호변환기 120에 입력되어 신호 변환된 심볼들로

출력된다. 상기 신호 변환기 120에서 이루어지는 신호 변환은 입력된 신호에 대해 0은 1로 변환하며, 1은 -1로 변환하여 출력하는 동작에 의해 이루어진다. 그러면, 상기 신호 변환된 심볼들은 확산기 130으로 입력되어, 확산되어진 후 출력되어진다.

상기 도 1에서 알 수 있듯이 데이터율 지시자는 4비트 또는 7비트로 표현되어지고, 전송되어지기 전에 24개의 부호화 심볼로 부호화되어진다. 이때, 상기 부호화 심볼로 부호화된 상기 데이터율 지시자가 전송 중 에러가 생길 경우, 이에 대응하는 역방향 패킷 채널의 부호율 및 심볼 반복 횟수, 대역 확산을 위한 알시 부호 등의 길이 및 종류 등을 잘못 가리키게 되어 수신기에서 역방향 패킷 채널의 올바른 해석이 불가능해진다. 따라서, 상기와 같은 데이터율 지시자는 우수한 성능을 가지는 (24,4) 또는 (24,7) 부호기로 부호화되어야 할 뿐만 아니라, 상기 대응하는 패킷채널을 해석하기 위해서는 최대한 빠르게 복호화 되어야 한다.

발명이 이루고자 하는 기술적 과제

따라서, 상기한 바와 같은 문제점을 해결하기 위한 본 발명의 목적은 최적의 성능을 가지는 데이터율 지시자 부호화 방법 및 장치를 제공함에 있다.

본 발명의 다른 목적은 최소의 복잡도를 가지는 데이터율 지시자 부호화 방법 및 장치를 제공함에 있다.

본 발명의 또 다른 목적은 확장된 일차 리드뮬러 부호를 천공하는 방법을 사용함으로써 복호화 과정에서 역하다마드를 사용할 수 있게 되어 하드웨어 복잡도(Hardware Complexity)를 최소화할 수 있는 방법 및 장치를 제공함에 있다.

본 발명은 또 다른 목적은 확장된 일차 리드뮬러 부호를 천공하는 방법을 사용함으로써 복호화 과정에서 역하다마드를 사용할 수 있게 되어 최적의 부호어를 사용할 수 있는 장치 및 방법을 제공함에 있다.

본 발명의 또 다른 목적은 천공하기 전의 직교부호의 길이를 최대한 줄임으로써 하드웨어 복잡도를 최소화할 수 있는 장치 및 방법을 제공함에 있다.

본 발명의 또 다른 목적은 확장된 직교부호 부호를 천공하여 하드웨어 복잡도를 최소화할 뿐만 아니라, 오류 정정 성능에 있어서도 최적인 부호를 생성할 수 있는 장치 및 방법을 제공함에 있다.

본 발명의 또 다른 목적은 하드웨어 복잡도를 최소화하고, 오류 정정 성능에 있어 최적의 부호를 생성하는 (24,1) 내지 (24,7) 부호화를 겸용하는 장치 및 방법을 제공함에 있다.

발명의 구성 및 작용

이하 본 발명의 일 실시 예를 첨부된 도면을 참조하여 상세히 설명하면 다음과 같다.

한편, 본 발명의 실시 예는 본 발명의 주된 내용을 구체화하기 위하여 필요한 것이며, 본 발명의 내용을 제한하지는 않는다. 또한, 본 발명의 실시 예를 설명함에 있어 앞에서 설명한 구성요소와 동일한 동작을 하는 다른 도면의 구성요소는 이전과 동일한 도면 참조번호를 사용하고 있음에 주의하여야 할 것이다.

선형 오류정정부호(Linear Error Correcting Code)의 성능을 나타내는 척도(measure)로서는 오류정정부호의 부호어(code word)의 해밍 거리(Hamming distance)의 분포가 있다. 이는 각각의 부호어에서 0이 아닌 심볼의 개수를 의미한다. 예컨대, 0111이 어떤 부호어라면 상기 부호어의 해밍거리는 상기 부호어에 포함된 1의 개수, 즉 3이 상기 부호어의 해밍거리가 된다. 이 때, 여러 부호어의 해밍거리 값들 중 가장 작은 값을 최소거리(d_{min} : minimum distance)라고 칭한다. 한편, 상기 오류정정부호의 오류정정 성능은 상기의 최소거리가 클수록 우수함은 이미 자명한 사실이다. 이

는 참조문헌 " The Theory of Error- Correcting Codes" - F.J.Macwilliams, N.J.A. Sloane, North- Holland에서 자세히 게시하고 있다. 또한, 최적부호(optimal code)가 되기 위한 이진 선형 부호의 입력과 출력값에 따른 부호간의 최소 거리는 참조문헌[1]에 나타나 있다.

**참조문헌[1] An Updated Table of Minimum- Distance Bounds for Binary Linear Codes (A.E. Brouwer and Tom Verhoeff, IEEE Transactions on Information Theory, VOL 39, NO. 2, MARCH 1993) **

도 10은 본 발명의 실시 예에 따른 데이터 지시자에 할당되는 비트 수를 최소화하기 위한 역방향 데이터를 지시채널의 송신기 구조를 나타내는 도면이다.

상기 도 10을 참조하여 설명하면, k비트의 데이터를 지시자는 부호기 1200에 입력되어진다. 상기 데이터를 지시자로 할당되는 비트수 k는 역방향 패킷 채널을 통해 단말이 전송할 수 있는 데이터율의 가지 수에 의하여 결정된다. 그러면, 부호기 1200은 상기 데이터를 부호화하여 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 출력하게 된다. 상기 부호기 1200은 입력되는 비트 수 k에 따라 (24,1)부호기, (24,2)부호기, (24,3)부호기, (24,4)부호기, (24,5)부호기, (24,6)부호기 및 (24,7)부호기 중 어느 하나의 부호기로 동작한다. 상기 부호기 1200으로부터의 부호화 심볼열은 심볼반복기 1210에 입력되고, 상기 심볼반복기 1210은 상기 입력되어진 24개의 부호화 심볼들을 16번 반복하여 출력한다. 상기 반복되어진 부호화 심볼들은 신호변환기 1220에 입력되어 신호 변환된 심볼들로 출력된다. 상기 신호변환기 1220에서 이루어지는 신호 변환은 입력된 신호에 대해 0은 1로 변환하며, 1은 - 1로 변환하여 출력하는 동작에 의해 이루어진다. 상기 신호 변환된 심볼들은 확산기 1230으로 입력되어, 확산되어진 후 출력되어진다.

전술한 바와 같이 데이터를 지시자로 할당되는 비트 수를 최소화시키는 것은 부호기의 부호화율을 낮게 함으로써 데이터를 지시채널의 성능을 향상시킬 수 있다.

이때, 상기 데이터를 지시자를 부호화하는 부호기는 (24,1)부호기, (24,2)부호기, (24,3)부호기, (24,4)부호기, (24,5)부호기, (24,6)부호기 및 (24,7)부호기이다. 상기 참조문헌[1]을 참조하면, 입력이 1비트이고 출력이 24인 최적의 (24,1)선형 부호기는 최소거리로 24를 가지고, 입력이 2비트이고 출력이 24인 최적의 (24,2)선형 부호기는 최소거리로 16를 가진다. 입력이 3비트이고 출력이 24인 최적의 (24,3)선형 부호기는 최소거리로 13을 가지고, 입력이 4비트이고 출력이 24인 최적의 (24,4)선형 부호기는 최소거리로 12를 가진다. 입력이 5비트이고 출력이 24인 최적의 (24,5)선형 부호기는 최소거리로 12를 가지고, 입력이 6비트이고 출력이 24인 최적의 (24,6)선형 부호기는 최소거리로 10을 가진다. 마지막으로, 입력이 7비트이고 출력이 24인 최적의 (24,7)선형 부호기는 최소거리로 10을 가진다.

먼저, 상기 데이터를 지시자를 부호화 하는 (24,1) 부호기에 대해 살펴보면 다음과 같다.

1. (24,1)부호기

우선 본 발명에서 제공하는 (24,1)부호기는 (2,1) 리드풀러 부호를 32번 반복하고, 40심볼을 천공함으로써 최적의 (24,1)부호를 얻도록 하는 구조이다. 실제로, 선형 (24,1)부호를 생성할 수 있는 방법은 무수히 많다. 하지만, 본 발명의 실시 예에 따른 일차 리드풀러 부호를 천공하는 방법을 사용함으로써 하드웨어 복잡도(Hardware Complexity)를 최소화 할 수 있을 뿐만 아니라 오류 정정 성능에 있어서도 최적의 부호어를 사용할 수 있다. 본 발명의 실시 예에서는 리드풀러 부호를 가지고 오류 정정 부호를 생성하기로 한다. 또한, 상기 천공하기 전의 리드풀러 부호의 길이를 최대한 줄임으로써 하드웨어 복잡도를 최소화할 수 있다.

상술한 바와 같이 (24,1) 부호기로부터 출력되는 부호어는 길이 (2,1) 리드밀러 부호 발생기에서 출력된 2개의 부호화 심볼들을 32번 반복한 후 40 심볼을 천공하여 총 길이 24의 부호화 심볼들을 출력한 것이다. 여기서, 상기 총 길이 64의 부호화 심벌들에서 상기 40 심볼을 천공하는 천공위치를 다르게 하면 부호어의 최소거리(d_{min} : minimum distance)가 상이하게 된다. 그러므로, 상기 총 길이 64의 부호화 심벌들에서 우수한 오류 정정 성능을 가지는 (24,1) 부호기를 만들기 위해서는 가장 큰 최소거리를 얻을 수 있는 천공위치를 구하는 것이 중요함을 나타내는 것이다.

상기와 같은 (24,1) 선형부호로써 최적의 부호를 생성하기 위해서 필요한 40개의 천공위치 중 가장 간단한 경우는 { 짹수 번째 부호화 심볼들, 1, 3, 5, 7, 9, 11, 13, 15} 이다. 이 때, 1개의 정보비트를 전송하기 위하여 본 발명의 부호화/복호화 방법을 사용하는 이동통신시스템의 송신기와 수신기는 미리 40개의 천공 위치를 약속하고 있어야 한다. 이는 통신 규격에서 정하여 두는 것이 일반적이며, 송신측이 미리 천공위치를 알려 주는 것도 가능하다.

도 11은 본 발명의 실시 예에 따른 송신기에 구비된 부호기의 구조를 도시하는 블록도이다. 상술한 본 발명에 따른 최적 (24,1) 부호의 부호화를 상기 도 11을 참조하여 설명하기로 한다.

상기 도 11를 참조하면, 1비트의 입력 정보비트(a_0)는 (2,1) 부호기(1300)의 리드밀러 부호기(1300)로 입력된다. 여기서, 상기 입력 정보비트는 데이터를 지시자이다. 상기 1비트의 입력 정보비트(a_0)를 입력받은 상기 리드밀러 부호기(1300)는 상기 1비트의 입력 정보비트(a_0)를 부호화하여 길이 2의 부호화 심볼들(부호화 심볼열)로 출력한다. 상기 부호화 심볼들은 리드밀러 부호이다. 상기 2개의 부호화 심볼들은 반복기 1310으로 입력된다. 상기 반복기 1310은 상기 2개의 입력 부호화 심볼들을 32회 반복하여 64개의 부호화 심볼들을 출력한다. 상기 64개의 부호화 심볼들은 천공기 1320으로 입력된다. 상기 천공기 1320은 상기 입력된 64개의 부호화 심볼들에 대해 40개 최적의 천공위치들의 부호화 심볼들을 천공하여 길이 24의 부호화 심볼들을 출력한다. 상기 40개 최적의 천공위치들은 이미 앞에서도 밝히고 있는 바와 같이 모든 짹수 번째(2,4,6,8,10,12,14,16,18,20,22,24,26,28,30,32,34,36,38,40,42,44,46,48,50,52,54,56,58,60,62,64 번째 심볼들)와 1,3,5,7,9,11,13,15 번째 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

일반적인 부호 이론에서는 입력정보와 부호화 심볼들간의 대응관계를 나타내기 위해서 생성행렬(Generating Matrix)을 정의한다. 상기와 같이 심볼 반복 및 천공을 포함하였을 때, 최종적인 (24,1) 부호기에 대한 생성행렬은 하기의 <수학식 1> 에서 나타내고 있는 바와 같다.

수학식 1

$$M = [1111 \ 1111 \ 1111 \ 1111 \ 1111 \ 1111]$$

상기 < 수학식 1> 에 나타난 생성행렬은 상기 1비트의 입력신호가 1이면 첫 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 선택하지 않을 경우 심볼열은 24개의 0으로 이루어지게 된다.

도 12은 상기 생성행렬에 따른 (24,1) 부호에 대한 부호기를 나타낸다.

상기 도 12를 참조하여 상기 부호기의 동작을 설명하면, 0 또는 1의 값을 가지는 입력 정보비트 a_0 은 승산기 1410으로 입력된다. 이와 동시에 상기 신호발생기 1400은 메모리에 저장되어 있는 상기 생성행렬을 구성하는 한 개의 심볼열을 출력한다. 즉, 상기 생성행렬의 열에 해당하는 길이 24인 심볼열 $R_1 = 1111 \ 1111 \ 1111 \ 1111 \ 1111 \ 1111$ 을 메모리에 저장하고 있다가 승산기 1410으로 출력한다. 그러면, 상기 승산기 1410은 상기 입력된 심볼열 R_1 의 각각의 심볼들과 상기 입력되어진 입력 정보비트 a_0 을 승산하여 길이 24인 부호화 심볼열을 출력한다.

2. (24,2) 부호기

우선 본 발명에서 제공하는 (24,2) 부호기는 (4,2) 리드룰러 부호를 8번 반복하고, 8심볼을 천공함으로써 최적의 (24,2) 부호를 얻도록 하는 구성이다. 실제로, 선형 (24,2) 부호를 생성할 수 있는 방법은 무수히 많지만 본 발명의 실시 예에 따른 일차 리드룰러 부호를 천공하는 방법을 사용함으로써 하드웨어 복잡도(Hardware Complexity)를 최소화 할 수 있을 뿐만 아니라 오류 정정 성능에 있어서도 최적인 최적 부호를 생성할 수 있다. 본 발명의 실시 예에서는 리드룰러 부호를 가지고 오류 정정 부호를 생성하기로 한다. 또한, 상기 천공하기 전의 리드룰러 부호의 길이를 최대한 줄임으로써 하드웨어 복잡도를 최소화 할 수 있다.

상술한 바와 같이 (24,2) 부호기로부터 출력되는 부호어는 길이 (4,2) 리드룰러 부호 발생기에서 출력된 4개의 부호화 심볼들을 8번 반복하고, 8심볼을 천공하여 총 길이 24의 부호화 심볼들을 출력한 것이다. 여기서, 상기 총 길이 32의 부호화 심벌들에서 상기 8 심볼을 천공하는 천공위치를 다르게 하면 부호어의 최소거리(d_{min} : minimum distance)가 상이하게 된다. 그러므로, 상기 총 길이 32의 부호화 심벌들에서 우수한 오류 정정 성능을 가지는 (24,2) 부호기를 만들기 위해서는 가장 큰 최소거리를 얻을 수 있는 천공위치를 구하는 것이 중요함을 나타내는 것이다.

상기와 같은 (24,2) 선형부호로써 최적의 부호를 생성하기 위해서 필요한 8개의 천공위치 중 가장 간단한 경우는 {0,4, 8,12,16,20,24,28} 이다. 이 때, 2개의 정보비트를 전송하기 위하여 본 발명의 부호화/복호화 방법을 사용하는 이동통신시스템의 송신기와 수신기는 미리 8개의 천공 위치를 약속하고 있어야 한다. 이는 통신 규격에서 정하여 두는 것이 일반적이며, 송신측이 미리 천공위치를 알려 주는 것도 가능하다.

상기에서 설명한 바와 같이 본 발명에 따른 최적 (24,2) 부호의 부호화를 도 13를 참조하여 설명하기로 한다. 상기 도 13는 본 발명의 실시 예에 따른 송신기에 구비된 부호기의 구조를 도시하는 블록도이다.

상기 도 13를 참조하면, 2비트의 입력 정보비트들(a_0, a_1)은 리드룰러 부호기에 해당하는 (4,2) 부호기(1500)로 입력된다. 여기서, 상기 입력 정보비트들은 데이터율 지시자이다. 상기 2비트의 입력 정보비트들(a_0, a_1)을 입력받은 상기 리드룰러 부호기(1500)는 상기 2비트의 입력 정보비트들(a_0, a_1)을 부호화하여 길이 4의 부호화 심볼들(부호 심볼열)로 출력한다. 상기 부호화 심볼들은 리드룰러 부호이다. 상기 4개의 부호화 심볼들은 반복기 1510으로 입력된다. 상기 반복기 1510은 상기 길이 4의 입력 부호화 심볼들을 8회 반복하여 32개의 부호화 심볼들을 출력하게 된다. 상기 32개의 부호화 심볼들이 천공기 1520으로 입력되어지면 상기 천공기 1520은 상기에서 나타나있는 8개의 최적의 천공위치인 모든 0,4,8,12,16,20,24,28 번째 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

일반적인 부호 이론에서는 입력정보와 부호화 심볼들간의 대응관계를 나타내기 위해서 생성행렬(Generating Matrix)을 정의한다. 상기와 같이 심볼 반복 및 천공을 포함하였을 때, 최종적인 (24,2) 부호기에 대한 생성행렬은 하기의 <수학식 2> 에서 나타내고 있는 바와 같다.

수학식 2

$$M = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

상기 < 수학식 2> 에 나타난 생성행렬은 상기 2비트의 입력 정보비트들 중 첫 번째 입력 정보비트가 1이면 첫 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 2비트의 입력 정보비트들 중 두 번째 입력 정보비트가 1이면 두 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 이때, 상기 선택되어진 모든 열들에 대해 심볼 단위로 배타적 가산을 하면, 상기 입력 정보비트에 대응되는 부호화 심볼열이 된다.

도 14은 상기 생성행렬에 따른 (24,2)부호에 대한 부호기를 나타낸다.

상기 도 14을 참조하여 상기 부호기의 동작을 설명하면, 0 또는 1의 값을 가지는 입력 정보비트 a_0, a_1 이 입력되면, a_0 은 승산기 1620으로, a_1 은 승산기 1622로 각각 입력된다. 이와 동시에 상기 신호발생기 1600은 메모리에 저장되어 있는 상기 생성행렬을 구성하는 두 개의 심볼열들을 상기 승산기들 중 각각 대응하는 하나의 승산기들로 출력한다. 즉, 상기 생성행렬의 첫 번째 열에 해당하는 길이 24인 심볼열 $R_1 = 101\ 101\ 101\ 101\ 101\ 101\ 101\ 101\ 101$ 을 메모리에 저장하고 있다가 승산기 1620으로 출력한다. 상기 생성행렬의 두 번째 열에 해당하는 길이 24인 심볼열 $R_2 = 011\ 011\ 011\ 011\ 011\ 011\ 011\ 011$ 을 메모리에 저장하고 있다가 승산기 1622로 출력한다. 그러면, 상기 승산기 1620은 상기 입력된 심볼열 R_1 의 각각의 심볼들과 상기 입력되어진 입력 정보비트 a_0 을 승산하여 길이 24인 심볼열을 배타적 가산기 1640으로 출력한다. 한편, 상기 승산기 1622는 상기 입력된 심볼열 R_2 의 각각의 심볼들과 상기 입력되어진 입력신호 a_1 을 승산하여 길이 24인 심볼열을 상기 배타적 가산기 1640으로 출력한다. 그러면, 상기 배타적 가산기 1640은 상기 입력된 4개의 길이 24인 심볼열들을 심볼 단위로 배타적 가산하여 길이 24인 부호화 심볼열을 출력한다.

3. (24,3) 부호기

우선 본 발명에서 제공하는 (24,3)부호기는 (8,3) 리드뮬러 부호를 4번 반복하고 8심볼을 천공함으로써 최적의 (24,3)부호를 얻도록 하는 구조이다. 실제로, 선형 (24,3)부호를 생성할 수 있는 방법은 무수히 많지만 본 발명의 실시 예에 따른 일차 리드뮬러 부호를 천공하는 방법을 사용함으로써 하드웨어 복잡도(Hardware Complexity)를 최소화 할 수 있을 뿐만 아니라 오류 정정 성능에 있어서도 최적인 최적 부호를 생성할 수 있다. 본 발명의 실시 예에서는 리드뮬러 부호를 가지고 오류 정정 부호를 생성하기로 한다. 또한, 상기 천공하기 전의 리드뮬러 부호의 길이를 최대한 줄임으로써 하드웨어 복잡도를 최소화할 수 있다.

상술한 바와 같이 (24,3)부호기로부터 출력되는 부호어는 길이 (8,3) 리드뮬러 부호 발생기에서 출력된 8개의 부호화 심볼들을 4번 반복하고 8심볼을 천공하여 총 길이 24의 부호화 심벌들을 출력한 것이다. 여기서, 상기 총 길이 32의 부호화 심벌들에서 상기 8 심볼을 천공하는 천공위치를 다르게 하면 부호어의 최소거리(d_{min} : minimum distance)가 상이하게 된다. 그러므로, 상기 총 길이 32의 부호화 심벌들에서 우수한 오류 정정 성능을 가지는 (24,3) 부호기를 만들기 위해서는 가장 큰 최소거리를 얻을 수 있는 천공위치를 구하는 것이 중요함을 나타내는 것이다.

상기와 같은 (24,3)선형부호로써 최적의 부호를 생성하기 위해서 필요한 8개의 천공위치 중 가장 간단한 경우는 {0,3,5,6,7,8,16,24}이다. 이 때, 3개의 정보비트를 전송하기 위하여 본 발명의 부호화/복호화 방법을 사용하는 이동통신시스템의 송신기와 수신기는 미리 8개의 천공 위치를 약속하고 있어야 한다. 이는 통신 규격에서 정하여 두는 것이 일반적이며, 송신측이 미리 천공위치를 알려 주는 것도 가능하다.

상기에서 설명한 바와 같이 본 발명에 따른 최적 (24,3) 부호의 부호화를 도 15를 참조하여 설명하기로 한다. 상기 도 15는 본 발명의 실시 예에 따른 송신기에 구비된 부호기의 구조를 도시하는 블록도이다.

상기 도 15를 참조하면, 3비트의 입력 정보비트들(a_0, a_1, a_2)은 리드뮬러 부호기에 해당하는 (8,3) 부호기(1700)로 입력된다. 여기서, 상기 입력 정보비트들은 데이터를 지시자이다. 상기 3비트의 입력 정보비트들(a_0, a_1, a_2)을 입력받은 상기 리드뮬러 부호기(1700)는 상기 3비트의 입력 정보비트들(a_0, a_1, a_2)을 부호화하여 길이 8의 부호화 심볼들(부호 심볼열)로 출력한다. 상기 부호화 심볼들은 리드뮬러 부호이다. 상기 8개의 부호화 심볼들은 반복기 1710으로 입력된다. 상기 반복기 1710은 상기 입력 부호화 심볼들을 4회 반복하여 32개의 부호화 심볼들을 출력하게 된다. 상기 32개의 부호화 심볼들이 천공기 1720으로 입력되어지면 상기 천공기 1720은 상기에서 나타나있는 8개의 최적의 천공 위치인 모든 0,3,5,6,7,8,16,24 번째 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

일반적인 부호 이론에서는 입력정보와 부호화 심볼들간의 대응관계를 나타내기 위해서 생성행렬(Generating Matrix)을 정의한다. 상기와 같이 심볼 반복 및 천공을 포함하였을 때, 최종적인 (24,3)부호기에 대한 생성행렬은 하기의 <수학식 3>에서 나타내고 있는 바와 같다.

수학식 3

$$M = \begin{bmatrix} 100101010110101011010101 \\ 010011001101100110110011 \\ 00100011100011110001111 \end{bmatrix}$$

상기 < 수학식 3>에 나타난 생성행렬은 상기 4비트의 입력신호들 중 첫 번째 입력신호가 1이면 첫 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 3비트의 입력신호들 중 두 번째 입력신호가 1이면 두 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 3비트의 입력신호들 중 세 번째 입력신호가 1이면 세 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 이때, 상기 선택되어진 모든 열들에 대해 심볼 단위로 배타적 가산을 하면, 상기 입력정보비트에 대응되는 부호화 심볼열이 된다.

도 16은 상기 생성행렬에 따른 (24,3)부호에 대한 부호기를 나타낸다.

상기 도 16을 참조하여 상기 부호기의 동작을 설명하면, 0 또는 1의 값을 가지는 입력 정보비트 a_0, a_1, a_2 이 입력되면, a_0 은 승산기 1820으로, a_1 은 승산기 1822로, a_2 는 승산기 1824로 각각 입력된다. 이와 동시에 상기 신호발생기 1800은 메모리에 저장되어 있는 상기 생성행렬을 구성하는 세 개의 심볼열들을 상기 승산기들 중 각각 대응하는 하나의 승산기들로 출력한다. 즉, 상기 생성행렬의 첫 번째 열에 해당하는 길이 24인 심볼열 $R_1 = 100\ 101\ 0101\ 101\ 0101\ 101\ 0101$ 을 메모리에 저장하고 있다가 승산기 1820으로 출력하고, 상기 생성행렬의 두 번째 열에 해당하는 길이 24인 심볼열 $R_2 = 010\ 011\ 0011\ 011\ 0011\ 011\ 0011$ 을 메모리에 저장하고 있다가 승산기 1822로 출력한다. 또한, 상기 생성행렬의 세 번째 열에 해당하는 길이 24인 심볼열 $R_3 = 001\ 000\ 1111\ 000\ 1111\ 000\ 1111$ 을 메모리에 저장하고 있다가 승산기 1824로 출력한다. 그러면, 상기 승산기 1820은 상기 입력된 심볼열 R_1 의 각각의 심볼들과 상기 입력되어진 입력 정보비트 a_0 을 승산하여 길이 24인 심볼열을 배타적 가산기 1840으로 출력하고, 상기 승산기 1822는 상기 입력된 심볼열 R_2 의 각각의 심볼들과 상기 입력되어진 입력신호 a_1 을 승산하여 길이 24인 심볼열을 상기 배타적 가산기 1840으로 출력한다. 또한, 상기 승산기 1824는 상기 입력된 심볼열 R_3 의 각각의 심볼들과 상기 입력되어진 입력신호 a_2 를 승산하여 길이 24인 심볼열을 상기 배타적 가산기 1840으로 출력한다. 그러면, 상기 배타적 가산기 1840은 상기 입력된 3개의 길이 24인 심볼열들을 심볼 단위로 배타적 가산하여 길이 24인 부호화 심볼열을 출력한다.

4. (24,4) 부호기

우선 본 발명에서 제공하는 (24,4)부호기는 (16,4) 리드뮬러 부호를 4번 반복하고 8심볼을 천공함으로써 최적의 (24,4)부호를 얻도록 하는 구조이다. 실제로, 선형 (24,4)부호를 생성할 수 있는 방법은 무수히 많지만 본 발명의 실시 예에 따른 일차 리드뮬러 부호를 천공하는 방법을 사용함으로써 하드웨어 복잡도(Hardware Complexity)를 최소화 할 수 있을 뿐만 아니라 오류 정정 성능에 있어서도 최적인 최적 부호를 생성할 수 있다. 본 발명의 실시 예에서는 리드뮬러 부호를 가지고 오류 정정 부호를 생성하기로 한다. 또한, 상기 천공하기 전의 리드뮬러 부호의 길이를 최대한 줄임으로써 하드웨어 복잡도를 최소화 할 수 있다.

상술한 바와 같이 (24,4) 부호기로부터 출력되는 부호어는 길이 (16,4) 리드풀러 부호 발생기에서 출력된 16개의 부호화 심볼들을 2번 반복하고 8심볼을 천공하여 총 길이 24의 부호 심볼들을 출력한 것이다. 여기서, 상기 총 길이 32의 부호 심벌들에서 상기 8 심볼을 천공하는 천공위치를 다르게 하면 부호어의 최소거리(d_{min} : minimum distance)가 상이하게 된다. 그러므로, 상기 총 길이 32의 부호 심벌들에서 우수한 오류 정정 성능을 가지는 (24,4) 부호기를 만들기 위해서는 가장 큰 최소거리를 얻을 수 있는 천공위치를 구하는 것이 중요함을 나타내는 것이다.

상기와 같은 (24,4) 선형부호로써 최적의 부호를 생성하기 위해서 필요한 8개의 천공위치 중 가장 간단한 경우는 {0,1,2,3,4,5,6,16} 이다. 이 때, 4개의 정보비트를 전송하기 위하여 본 발명의 부호화/복호화 방법을 사용하는 이동통신시스템의 송신기와 수신기는 미리 8개의 천공 위치를 약속하고 있어야 한다. 이는 통신 규격에서 정하여 두는 것이 일반적이며, 송신측이 미리 천공위치를 알려 주는 것도 가능하다.

상기에서 설명한 바와 같이 본 발명에 따른 최적 (24,4) 부호의 부호화를 도 2를 참조하여 설명하기로 한다. 상기 도 2는 본 발명의 실시 예에 따른 송신기에 구비된 부호기의 구조를 도시하는 블록도이다.

상기 도 2를 참조하면, 4비트의 입력 정보비트들(a_0, a_1, a_2, a_3)은 (16,4) 부호기(200)의 리드풀러 부호기(200)로 입력된다. 여기서, 상기 입력 정보비트들은 데이터율 지시자이다. 상기 4비트의 입력 정보비트들(a_0, a_1, a_2, a_3)을 입력받은 상기 리드풀러 부호기(200)는 상기 4비트의 입력 정보비트들(a_0, a_1, a_2, a_3)을 부호화하여 길이 16의 부호화 심볼들(부호 심볼열)로 출력한다. 상기 부호화 심볼들은 리드풀러 부호이다. 상기 16개의 부호화 심볼들은 반복기 210으로 입력된다. 상기 반복기 210은 상기 입력 부호화 심볼들을 2회 반복하여 32개의 부호화 심볼들을 출력하게 된다. 그러면, 상기 32개의 부호화 심볼들이 천공기 220으로 입력되어야 하면 상기 천공기 220은 상기에서 나타나 있는 8개의 최적의 천공위치인 모든 0,1,2,3,4,5,6,16 번째 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

일반적인 부호 이론에서는 입력정보와 부호화 심볼들간의 대응관계를 나타내기 위해서 생성행렬(Generating Matrix)을 정의한다. 상기와 같이 심볼 반복 및 천공을 포함하였을 때, 최종적인 (24,4) 부호기에 대한 생성행렬은 하기의 <수학식 4>에서 나타내고 있는 바와 같다.

수학식 4

$$M = \begin{bmatrix} 10101010110101010101010101 \\ 100110011011001100110011 \\ 100001111000111100001111 \\ 011111110000000111111111 \end{bmatrix}$$

상기 < 수학식 4> 에 나타난 생성행렬은 상기 4비트의 입력신호들 중 첫 번째 입력신호가 1이면 첫 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 4비트의 입력신호들 중 두 번째 입력신호가 1이면 두 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 4비트의 입력신호들 중 세 번째 입력신호가 1이면 세 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 4비트의 입력신호들 중 네 번째 입력신호가 1이면 네 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 이때, 상기 선택되어진 모든 열들에 대해 심볼 단위로 배타적 가산을 하면, 상기 입력정보비트에 대응되는 부호화 심볼열이 된다.

도 7은 상기 생성행렬에 따른 (24,4) 부호에 대한 부호기를 나타낸다.

상기 도 7을 참조하여 상기 부호기의 동작을 설명하면, 0 또는 1의 값을 가지는 입력 정보비트 a0, a1, a2, a3이 입력되면, a0은 승산기 920으로, a1은 승산기 922로, a2는 승산기 924로, a3은 승산기 926으로 각각 입력된다. 이와 동시에 상기 신호발생기 900은 메모리에 저장되어 있는 상기 생성행렬을 구성하는 네 개의 심볼열들을 상기 승산기들 중 각각 대응하는 하나의 승산기들로 출력한다. 즉, 상기 생성행렬의 첫 번째 열에 해당하는 길이 24인 심볼열 R1 = 1010 1010 1101 0101 0101 0101을 메모리에 저장하고 있다가 승산기 920으로 출력하고, 상기 생성행렬의 두 번째 열에 해당하는 길이 24인 심볼열 R2 = 1001 1001 1011 0011 0011 0011을 메모리에 저장하고 있다가 승산기 922로 출력한다. 또한, 상기 생성행렬의 세 번째 열에 해당하는 길이 24인 심볼열 R3 = 1000 0111 1000 1111 0000 1111을 메모리에 저장하고 있다가 승산기 924로 출력하고, 상기 생성행렬의 네 번째 열에 해당하는 길이 24인 심볼열 R4 = 0111 1111 1000 0000 1111 1111을 메모리에 저장하고 있다가 승산기 926으로 출력한다. 그러면, 상기 승산기 920은 상기 입력된 심볼열 R1의 각각의 심볼들과 상기 입력되어진 입력 정보비트 a0을 승산하여 길이 24인 심볼열을 배타적 가산기 940으로 출력하고, 상기 승산기 922는 상기 입력된 심볼열 R2의 각각의 심볼들과 상기 입력되어진 입력 신호 a1을 승산하여 길이 24인 심볼열을 상기 배타적 가산기 940으로 출력한다. 또한, 상기 승산기 924는 상기 입력된 심볼열 R3의 각각의 심볼들과 상기 입력되어진 입력신호 a2를 승산하여 길이 24인 심볼열을 상기 배타적 가산기 940으로 출력하고, 상기 승산기 926은 상기 입력된 심볼열 R4의 각각의 심볼들과 상기 입력되어진 입력신호 a3을 승산하여 길이 24인 심볼열을 상기 배타적 가산기 940으로 출력한다. 그러면, 상기 배타적 가산기 940은 상기 입력된 4개의 길이 24인 심볼열들을 심볼 단위로 배타적 가산하여 길이 24인 부호화 심볼열을 출력한다.

5. (24,5) 부호기

우선 본 발명에서 제공하는 (24,5) 부호기는 (32,5) 일차 리드밀러 부호를 8심볼을 천공함으로써 최적의 (24,5) 부호를 얻도록 하는 구조이다. 실제로, 선형 (24,5) 부호를 생성할 수 있는 방법은 무수히 많지만 본 발명의 실시 예에 따른 일차 리드밀러 부호를 천공하는 방법을 사용함으로써 하드웨어 복잡도(Hardware Complexity)를 최소화 할 수 있을 뿐만 아니라 오류 정정 성능에 있어서도 최적인 최적 부호를 생성할 수 있다. 본 발명의 실시 예에서는 리드밀러 부호를 가지고 오류 정정 부호를 생성하기로 한다. 또한, 상기 천공하기 전의 일차 리드밀러 부호의 길이를 최대한 줄임으로써 하드웨어 복잡도를 최소화 할 수 있다.

상술한 바와 같이 (24,5) 부호기로부터 출력되는 부호어는 길이 (32,5) 리드밀러 부호 발생기에서 출력된 32개의 부호화 심볼들을 출력하고, 상기 리드밀러 부호 심볼들 중 8심볼을 천공한 것이다. 여기서, 상기 총 길이 32의 일차 리드밀러 부호 심벌들에서 상기 8 심볼을 천공하는 천공위치를 다르게 하면 부호어의 최소거리(dmin: minimum distance)가 상이하게 된다. 그러므로, 상기 (32,5) 일차 리드밀러 부호에서 우수한 오류 정정 성능을 가지는 (24,5) 부호기를 만들기 위해서는 가장 큰 최소거리를 얻을 수 있는 천공위치를 구하는 것이 중요함을 나타내는 것이다.

상기와 같은 (24,5) 선형부호로써 최적의 부호를 생성하기 위해서 필요한 8개의 천공위치 중 가장 간단한 경우는 {0,1,2,3,4,5,6,7}이다. 이 때, 5개의 정보비트를 전송하기 위하여 본 발명의 부호화/복호화 방법을 사용하는 이동통신시스템의 송신기와 수신기는 미리 8개의 천공 위치를 약속하고 있어야 한다. 이는 통신 규격에서 정하여 두는 것이 일반적이며, 송신측이 미리 천공위치를 알려 주는 것도 가능하다.

상기에서 설명한 바와 같이 본 발명에 따른 최적 (24,5) 부호의 부호화를 도 17를 참조하여 설명하기로 한다. 상기 도 17는 본 발명의 실시 예에 따른 송신기에 구비된 부호기의 구조를 도시하는 블록도이다.

상기 도 17를 참조하면, 5비트의 입력 정보비트들(a0,a1,a2,a3,a4)은 (32,5) 부호기(1900)의 일차 리드밀러 부호기(1900)로 입력된다. 여기서, 상기 입력 정보비트들은 데이터를 지시자이다. 상기 5비트의 입력 정보비트들(a0,a1,a2,a3,a4)을 입력받은 상기 리드밀러 부호기(1900)는 상기 5비트의 입력 정보비트들(a0,a1,a2,a3,a4)을 부호화하여 길

이 32의 부호화 심볼들(부호 심볼열)로 출력한다. 상기 부호화 심볼들은 일차 리드룰러 부호이다. 상기 32개의 부호화 심볼들이 천공기 1920으로 입력되어지면 상기 천공기 1920은 상기에서 나타나있는 8개의 최적의 천공위치인 0,1,2, 3,4,5,6,7번째 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

일반적인 부호 이론에서는 입력정보와 부호화 심볼들간의 대응관계를 나타내기 위해서 생성행렬(Generating Matrix)을 정의한다. 상기와 같이 심볼 반복 및 천공을 포함하였을 때, 최종적인 (24,5) 부호기에 대한 생성행렬은 하기의 <수학식 5>에서 나타내고 있는 바와 같다.

수학식 5

$$M = \begin{bmatrix} 01010101010101010101010101 \\ 001100110011001100110011 \\ 000011110000111100001111 \\ 111111110000000011111111 \\ 000000001111111111111111 \end{bmatrix}$$

상기 < 수학식 5> 에 나타난 생성행렬은 상기 5비트의 입력신호들 중 첫 번째 입력신호가 1이면 첫 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 5비트의 입력신호들 중 두 번째 입력신호가 1이면 두 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 5비트의 입력신호들 중 세 번째 입력신호가 1이면 세 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 5비트의 입력신호들 중 네 번째 입력신호가 1이면 네 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 5비트의 입력신호들 중 다섯 번째 입력신호가 1이면 네 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 이때, 상기 선택되어진 모든 열들에 대해 심볼 단위로 배타적 가산을 하면, 상기 입력정보비트에 대응되는 부호화 심볼열이 된다.

도 18은 상기 생성행렬에 따른 (24,5)부호에 대한 부호기를 나타낸다.

상기 도 18을 참조하여 상기 부호기의 동작을 설명하면, 0 또는 1의 값을 가지는 입력 정보비트 a_0, a_1, a_2, a_3, a_4 가 입력되면, a_0 은 승산기 2020으로, a_1 은 승산기 2022로, a_2 는 승산기 2024로, a_3 은 승산기 2026, a_4 는 승산기 2028로 각각 입력된다. 이와 동시에 상기 신호발생기 2000은 메모리에 저장되어 있는 상기 생성행렬을 구성하는 다섯 개의 심볼열들을 상기 승산기들 중 각각 대응하는 하나의 승산기들로 출력한다. 즉, 상기 생성행렬의 첫 번째 열에 해당하는 길이 24인 심볼열 $R_1 = 0101 0101 0101 0101 0101 0101$ 을 메모리에 저장하고 있다가 승산기 2020으로 출력하고, 상기 생성행렬의 두 번째 열에 해당하는 길이 24인 심볼열 $R_2 = 0011 0011 0011 0011 0011 0011$ 을 메모리에 저장하고 있다가 승산기 2022로 출력한다. 또한, 상기 생성행렬의 세 번째 열에 해당하는 길이 24인 심볼열 $R_3 = 0000 1111 0000 1111 0000 1111$ 을 메모리에 저장하고 있다가 승산기 2024로 출력하고, 상기 생성행렬의 네 번째 열에 해당하는 길이 24인 심볼열 $R_4 = 1111 1111 0000 0000 1111 1111$ 을 메모리에 저장하고 있다가 승산기 2026으로 출력하고, 상기 생성행렬의 다섯 번째 열에 해당하는 길이 24인 심볼열 $R_5 = 0000 0000 1111 1111 1111 1111$ 을 메모리에 저장하고 있다가 승산기 2028로 출력한다. 그러면, 상기 승산기 2020은 상기 입력된 심볼열 R_1 의 각각의 심볼들과 상기 입력되어진 입력 정보비트 a_0 을 승산하여 길이 24인 심볼열을 배타적 가산기 2040으로 출력하고, 상기 승산기 2022는 상기 입력된 심볼열 R_2 의 각각의 심볼들과 상기 입력되어진 입력신호 a_1 을 승산하여 길이 24인 심볼열을 상기 배타적 가산기 2040으로 출력한다. 또한, 상기 승산기 2024는 상기 입력된 심볼열 R_3 의 각각의 심볼들과 상기 입력되어진 입력신호 a_2 를 승산하여 길이 24인 심볼열을 상기 배타적 가산기 2040으로 출력하고, 상기 승산기 926은 상기 입력된 심볼열 R_4 의 각각의 심볼들과 상기 입력되어진 입력신호 a_3 을 승산하여 길이 24인 심볼열을 상기 배타적 가산기 2040으로 출력하고, 상기 승산기 2028은 상기 입력된 심볼열 R_5 의 각각의 심볼들과 상기 입력되어진 입력신호 a_4 을 승산하여 길이 24인 심볼열을 상기 배타적 가산기 2040으로 출력한다. 그러면, 상기 배타적 가산기 2040은

상기 입력된 5개의 길이 24인 심볼열들을 심볼 단위로 배타적 가산하여 길이 24인 부호화 심볼열을 출력한다.

6. (24,6) 부호기

우선 본 발명에서 제공하는 (24,6) 부호기는 (32,5) 직교부호(일차 리드뮐러 부호- First order Reed Muller code)에 1개의 마스크(mask) 함수를 사용하여 부호어를 확장한 확장된 직교부호에서 8심볼을 천공함으로써 최적의 (24,6) 부호를 얻도록 하는 구성이다. 이 때, 상기 확장된 직교부호의 구조는 도 19에서 도시되어진다.

상기 도 19을 참조하여 확장된 직교부호의 구조를 설명하면, M1이 상기 사용되어지는 1개의 마스크함수라 할 때, 상위 32개의 부호어는 상기 32개의 길이 32인 직교부호어들(W)을 사용한다. 다음 32개의 부호어는 상기 마스크 함수 M1과 상기 32개의 직교부호어들(W)을 각각 배타적 가산한 32개의 부호어들($M1 + W$)을 사용한다. 따라서, 확장된 직교부호로는 총 개수 $2^6 = 64$ 개의 부호어를 사용하게 된다. 이 때, (24,6) 부호로 최적이 되기 위한 상기 마스크함수는 실험적으로 찾을 수 있다.

예컨대, 상기와 같은 마스크 함수 M1는 다음과 같다.

$M1 = 0000\ 0000\ 1110\ 1000\ 1101\ 1000\ 1100\ 0000$

일반적으로, 선형 (24,6) 부호를 생성할 수 있는 방법은 무수히 많지만 본 발명의 실시 예에 따른 확장된 일차 리드뮐러 부호를 천공하는 방법을 사용함으로써 복호화 과정에서 역마다드를 사용할 수 있게 되어 하드웨어 복잡도(Hardware Complexity)를 최소화 할 수 있을 뿐만 아니라 오류 정정 성능에 있어서도 최적인 최적 부호를 생성할 수 있다. 본 발명의 실시 예에서는 확장된 직교부호를 가지고 오류 정정 부호를 생성하기로 한다. 또한, 상기 천공하기 전의 직교부호의 길이를 최대한 줄임으로써 하드웨어 복잡도를 최소화 할 수 있다.

상술한 바와 같이 (24,6) 부호어는 길이 (32,6) 확장된 직교부호 발생기에서 출력된 32개의 부호화 심볼들 중 8심볼을 천공한 것이다. 여기서, 상기 32개의 확장된 직교부호 심벌들 중에서 상기 8 심볼을 천공하는 천공위치를 다르게 하면 부호어의 최소거리(dmin: minimum distance)가 상이하게 되는 것이다. 그러므로, 상기 (32,6) 확장된 직교부호에서 우수한 오류 정정 성능을 가지는 (24,6) 부호기를 만들기 위해서는 가장 큰 최소거리를 얻을 수 있도록 하는 천공위치를 구할 수 있어야 한다.

상기와 같은 (24,6) 선형부호로써 최적의 부호를 생성하기 위해서 필요한 8개의 천공위치는 실험적으로 구할 수 있는데, 그 중 가장 간단한 경우는 { 0,1,2,3,4,5,6,7 } 이다. 이 때, 6개의 정보비트를 전송하기 위하여 본 발명의 부호화/복호화 방법을 사용하는 이동통신시스템의 송신기와 수신기는 미리 8개의 천공 위치를 약속하고 있어야 한다. 이는 통신 규격에서 정하여 두는 것이 일반적이며, 송신측이 미리 천공위치를 알려 주는 것도 가능하다.

상기에서 설명한 바와 같이 본 발명의 실시 예에 따른 송신기에 구비된 최적 (24,6) 부호기의 구성은 도 20에서 보이고 있는 바와 같다.

상기 도 20을 참조하면, 6비트의 입력 정보비트들(a0,a1,a2,a3,a4,a5)은 확장된 직교부호기에 해당하는 (32,6) 부호기(2100)로 입력된다. 상기 6비트의 입력 정보비트들은 데이터를 지시자들이다. 상기 6비트의 입력 정보비트들(a0,a1,a2,a3,a4,a5)을 입력받은 상기 확장된 직교부호기(2100)는 상기 6비트의 입력 정보비트들(a0,a1,a2,a3,a4,a5)을 부호화하여 길이 32의 부호화 심볼들(부호 심볼열)을 출력한다. 상기 확장된 직교부호기(2100)로부터 출력되는 상기 32개의 부호화 심볼들은 천공기 2110으로 입력된다. 상기 천공기 2110은 상기에서 나타나있는 8개의 최적의 천공위치인 0,1,2,3,4,5,6,7번째 심볼들에 의해 상기 32개의 부호화 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된

다.

일반적인 부호 이론에서는 입력정보와 부호화 심볼들간의 대응관계를 나타내기 위해서 생성행렬(Generating Matrix)을 정의한다. 상기와 같이 심볼 천공을 포함하였을 때, 최종적인 (24,6) 부호기에 대한 생성행렬은 하기의 < 수학식 6>에서 나타내고 있는 바와 같다.

수학식 6

$$M = \begin{bmatrix} 010101010101010101010101 \\ 001100110011001100110011 \\ 000011110000111100001111 \\ 111111110000000011111111 \\ 000000001111111111111111 \\ 111010001101100011000000 \end{bmatrix}$$

상기 < 수학식 6> 에 나타난 생성행렬은 상기 6비트의 입력신호들 중 첫 번째 입력신호가 1이면 첫 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 6비트의 입력신호들 중 두 번째 입력신호가 1이면 두 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 6비트의 입력신호들 중 세 번째 입력신호가 1이면 세 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 6비트의 입력신호들 중 네 번째 입력신호가 1이면 네 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 6비트의 입력신호들 중 다섯 번째 입력신호가 1이면 다섯 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 6비트의 입력신호들 중 여섯 번째 입력신호가 1이면 여섯 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 이때, 상기 선택되어진 모든 열들에 대해 심볼 단위로 배타적 가산을 하면, 상기 입력정보비트에 대응되는 부호화 심볼열이 된다.

도 21은 상기 생성행렬에 따른 (24,6) 부호에 대한 부호기를 나타낸다.

상기 도 21을 참조하여 상기 부호기의 동작을 설명하면, 0 또는 1의 값을 가지는 입력 정보비트 $a_0, a_1, a_2, a_3, a_4, a_5$ 가 입력되면, a_0 은 승산기 2220으로, a_1 은 승산기 2222로, a_2 는 승산기 2224로, a_3 은 승산기 2226으로, a_4 는 승산기 2228로, a_5 은 승산기 2230으로 각각 입력된다. 이와 동시에 상기 신호발생기 2200은 메모리에 저장되어 있는 상기 생성행렬을 구성하는 여섯 개의 심볼열들을 상기 승산기들 중 각각 대응하는 하나의 승산기들로 출력한다. 즉, 상기 생성행렬의 첫 번째 열에 해당하는 길이 24인 심볼열 $R_1 = 0101 0101 0101 0101 0101 0101$ 을 메모리에 저장하고 있다가 승산기 2220으로 출력하고, 상기 생성행렬의 두 번째 열에 해당하는 길이 24인 심볼열 $R_2 = 0011 001 1 0011 0011 0011$ 을 메모리에 저장하고 있다가 승산기 2222로 출력한다. 또한, 상기 생성행렬의 세 번째 열에 해당하는 길이 24인 심볼열 $R_3 = 0000 1111 0000 1111 0000 1111$ 을 메모리에 저장하고 있다가 승산기 2224로 출력하고, 상기 생성행렬의 네 번째 열에 해당하는 길이 24인 심볼열 $R_4 = 1111 1111 0000 0000 1111 1111$ 을 메모리에 저장하고 있다가 승산기 2226으로 출력한다. 상기 생성행렬의 다섯 번째 열에 해당하는 길이 24인 심볼열 $R_5 = 0000 0000 1111 1111 1111 1111$ 을 메모리에 저장하고 있다가 승산기 2228로 출력하고, 상기 생성행렬의 여섯 번째 열에 해당하는 길이 24인 심볼열 $R_6 = 1110 1000 1101 1000 1100 0000$ 을 메모리에 저장하고 있다가 승산기 2230으로 출력한다. 그러면, 상기 승산기 2220은 상기 입력된 심볼열 R_1 의 각각의 심볼들과 상기 입력되어진 입력신호 a_0 을 승산하여 길이 24인 심볼열을 배타적 가산기 2240으로 출력하고, 상기 승산기 2222는 상기 입력된 심볼열 R_2 의 각각의 심볼들과 상기 입력되어진 입력신호 a_1 을 승산하여 길이 24인 심볼열을 상기 배타적 가산기 2240으로 출력한다. 상기 승산기 2224는 상기 입력된 심볼열 R_3 의 각각의 심볼들과 상기 입력되어진 입력신호 a_2 를 승산하여 길이 24인 심볼열을 상기 배타적 가산기 2240으로 출력하고, 상기 승산기 1026은 상기 입력된 심볼열 R_4 의 각각의 심볼

들과 상기 입력되어진 입력신호 a3을 승산하여 길이 24인 심볼열을 상기 배타적 가산기 2240으로 출력한다. 상기 승산기 2228은 상기 입력된 심볼열 R5의 각각의 심볼들과 상기 입력되어진 입력신호 a4를 승산하여 길이 24인 심볼열을 상기 배타적 가산기 2240으로 출력하고, 상기 승산기 2230은 상기 입력된 심볼열 R6의 각각의 심볼들과 상기 입력되어진 입력신호 a5를 승산하여 길이 24인 심볼열을 상기 배타적 가산기 1040으로 출력한다. 그러면, 상기 배타적 가산기 2240은 상기 입력된 6개의 길이 24인 심볼열들을 심볼 단위로 배타적 가산하여 길이 24인 부호화 심볼열을 출력한다.

7. (24,7) 부호기

우선, 본 발명에서 제공하는 (24,7) 부호기는 (32,5) 직교부호(일차 리드뮬러 부호- First order Reed Muller code)에 2개의 마스크(mask) 함수를 사용하여 부호어를 확장한 확장된 직교부호에서 8심볼을 천공함으로써 최적의 (24,7) 부호를 얻도록 하는 구성이다. 이 때, 상기 확장된 직교부호의 구조는 도 3에서 도시되어진다.

상기 도 3을 참조하여 확장된 직교부호의 구조를 설명하면, M1과 M2가 상기 사용되어지는 2개의 마스크함수라 할 때, 상위 32개의 부호어는 상기 32개의 길이 32인 직교부호어들(W)을 사용한다. 다음 32개의 부호어는 상기 마스크 함수 M1과 상기 32개의 직교부호어들(W)을 각각 배타적 가산한 32개의 부호어들(M1+ W)을 사용하고, 그 다음 32개의 부호어는 상기 마스크 함수 M2와 상기 32개의 직교부호어들(W)을 각각 배타적 가산한 32개의 부호어들(M2+ W)을 사용한다. 마지막 32개의 부호어로는 상기 마스크 함수 M1,M2와 상기 32개의 직교부호어들(W)을 각각 배타적 가산한 32개의 부호어들(M1+ M2+ W)을 사용한다. 따라서, 확장된 직교부호로는 총 개수 $2^7 = 128$ 개의 부호어를 사용하게 된다. 이 때, (24,7) 부호로 최적이 되기 위한 상기 2개의 마스크함수는 실험적으로 찾을 수 있다.

예컨대, 상기와 같은 2개의 마스크 함수 M1, M2는 다음과 같다.

M1 = 0000 0000 1110 1000 1101 1000 1100 0000

M2 = 0000 0000 1100 0000 0111 1110 0010 1000

일반적으로, 선형 (24,7) 부호를 생성할 수 있는 방법은 무수히 많지만 본 발명의 실시 예에 따른 확장된 일차 리드뮬러 부호를 천공하는 방법을 사용함으로써 복호화 과정에서 역하다마드를 사용할 수 있게 되어 하드웨어 복잡도(Hardware Complexity)를 최소화 할 수 있을 뿐만 아니라 오류 정정 성능에 있어서도 최적인 최적 부호를 생성할 수 있다. 본 발명의 실시 예에서는 확장된 직교부호를 가지고 오류 정정 부호를 생성하기로 한다. 또한, 상기 천공하기 전의 직교부호의 길이를 최대한 줄임으로써 하드웨어 복잡도를 최소화할 수 있다.

상술한 바와 같이 (24,7) 부호어는 길이 (32,7) 확장된 직교부호 발생기에서 출력된 32개의 부호화 심볼들 중 8심볼을 천공한 것이다. 여기서, 상기 32개의 확장된 직교부호 심벌들 중에서 상기 8 심볼을 천공하는 천공위치를 다르게 하면 부호어의 최소거리(dmin: minimum distance)가 상이하게 되는 것이다. 그러므로, 상기 (32,7) 확장된 직교부호에서 우수한 오류 정정 성능을 가지는 (24,7) 부호기를 만들기 위해서는 가장 큰 최소거리를 얻을 수 있도록 하는 천공위치를 구할 수 있어야 한다.

상기와 같은 (24,7) 선형부호로써 최적의 부호를 생성하기 위해서 필요한 8개의 천공위치는 실험적으로 구할 수 있는데, 그 중 가장 간단한 경우는 {0,4,8,12,16,20,24,28} 또는 {0,1,2,3,4,5,6,7} 이다. 이 때, 7개의 정보비트를 전송하기 위하여 본 발명의 부호화/복호화 방법을 사용하는 이동통신시스템의 송신기와 수신기는 미리 8개의 천공 위치를 약속하고 있어야 한다. 이는 통신 규격에서 정하여 두는 것이 일반적이며, 송신측이 미리 천공위치를 알려 주는 것도 가능하다.

상기에서 설명한 바와 같이 본 발명의 실시 예에 따른 송신기에 구비된 최적 (24,7) 부호기의 구성은 도 4에서 보이고 있는 바와 같다.

상기 도 4를 참조하면, 7비트의 입력 정보비트들(a0,a1,a2,a3,a4,a5,a6)은 확장된 직교부호기에 해당하는 (32,7) 부호기(400)로 입력된다. 상기 7비트의 입력 정보비트들은 데이터를 지시자들이다. 상기 7비트의 입력 정보비트들(a0,a1,a2,a3,a4,a5,a6)을 입력받은 상기 확장된 직교부호기(400)는 상기 7비트의 입력 정보비트들(a0,a1,a2,a3,a4,a5,a6)을 부호화하여 길이 32의 부호화 심볼들(부호 심볼열)을 출력한다. 상기 확장된 직교부호기(400)로부터 출력되는 상기 32개의 부호화 심볼들은 천공기 410으로 입력된다. 상기 천공기 410은 상기에서 나타나있는 8개의 최적의 천공 위치인 0,4,8,12,16,20,24,28번째 심볼들 또는 {0,1,2,3,4,5,6,7}에 의해 상기 32개의 부호화 심볼들을 천공하여 2 4개의 부호화 심볼들을 출력하게 된다.

일반적인 부호 이론에서는 입력정보와 부호화 심볼들간의 대응관계를 나타내기 위해서 생성행렬(Generating Matrix)을 정의한다. 상기와 같이 심볼 천공을 포함하였을 때, 최종적인 (24,7) 부호기에 대한 생성행렬은 하기의 < 수학식 7>에서 나타내고 있는 바와 같다.

수학식 7

$$M = \begin{bmatrix} 010101010101010101010101 \\ 001100110011001100110011 \\ 000011110000111100001111 \\ 111111110000000011111111 \\ 000000001111111111111111 \\ 111010001101100011000000 \\ 11000000011111000101000 \end{bmatrix}$$

상기 < 수학식 7>에 나타난 생성행렬은 상기 7비트의 입력신호들 중 첫 번째 입력신호가 1이면 첫 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 7비트의 입력신호들 중 두 번째 입력신호가 1이면 두 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 7비트의 입력신호들 중 세 번째 입력신호가 1이면 세 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 7비트의 입력신호들 중 네 번째 입력신호가 1이면 네 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 7비트의 입력신호들 중 다섯 번째 입력신호가 1이면 다섯 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 7비트의 입력신호들 중 여섯 번째 입력신호가 1이면 여섯 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 상기 7비트의 입력신호들 중 일곱 번째 입력신호가 1이면 일곱 번째 행의 24개의 심볼들을 선택하고, 0이면 선택하지 않는다. 이때, 상기 선택되어진 모든 열들에 대해 심볼 단위로 배타적 가산을 하면, 상기 입력정보비트에 대응되는 부호화 심볼열이 된다.

도 8은 상기 생성행렬에 따른 (24,7) 부호에 대한 부호기를 나타낸다.

상기 도 8을 참조하여 상기 부호기의 동작을 설명하면, 0 또는 1의 값을 가지는 입력 정보비트 a0, a1, a2, a3, a4, a5, a6이 입력되면, a0은 승산기 1020으로, a1은 승산기 1022로, a2는 승산기 1024로, a3은 승산기 1026으로, a4는 승산기 1028로, a5은 승산기 1030으로, a6은 승산기 1032로 각각 입력된다. 이와 동시에 상기 신호발생기 1000은 메모리에 저장되어 있는 상기 생성행렬을 구성하는 일곱 개의 심볼열들을 상기 승산기들 중 각각 대응하는 하나의 승산기들로 출력한다. 즉, 상기 생성행렬의 첫 번째 열에 해당하는 길이 24인 심볼열 R1 = 0101 0101 0101 0101 0101 0101을 메모리에 저장하고 있다가 승산기 1020으로 출력하고, 상기 생성행렬의 두 번째 열에 해당하는 길이 24인 심

불열 R2 = 0011 0011 0011 0011 0011 0011을 메모리에 저장하고 있다가 승산기 1022로 출력한다. 또한, 상기 생성행렬의 세 번째 열에 해당하는 길이 24인 심불열 R3 = 0000 1111 0000 1111 0000 1111을 메모리에 저장하고 있다가 승산기 1024로 출력하고, 상기 생성행렬의 네 번째 열에 해당하는 길이 24인 심불열 R4 = 1111 1111 0000 0000 1111 1111을 메모리에 저장하고 있다가 승산기 926으로 출력한다. 상기 생성행렬의 다섯 번째 열에 해당하는 길이 24인 심불열 R5 = 0000 0000 1111 1111 1111 1111을 메모리에 저장하고 있다가 승산기 1028로 출력하고, 상기 생성행렬의 여섯 번째 열에 해당하는 길이 24인 심불열 R6 = 1110 1000 1101 1000 1100 0000을 메모리에 저장하고 있다가 승산기 1030으로 출력한다. 마지막으로, 상기 생성행렬의 일곱 번째 열에 해당하는 길이 24인 심불열 R7 = 1100 0000 0111 1110 0010 1000을 메모리에 저장하고 있다가 승산기 1032로 출력한다. 그러면, 상기 승산기 1020은 상기 입력된 심불열 R1의 각각의 심불들과 상기 입력되어진 입력신호 a0을 승산하여 길이 24인 심불열을 배타적 가산기 1040으로 출력하고, 상기 승산기 1022는 상기 입력된 심불열 R2의 각각의 심불들과 상기 입력되어진 입력신호 a1을 승산하여 길이 24인 심불열을 상기 배타적 가산기 1040으로 출력한다. 상기 승산기 1024는 상기 입력된 심불열 R3의 각각의 심불들과 상기 입력되어진 입력신호 a2를 승산하여 길이 24인 심불열을 상기 배타적 가산기 1040으로 출력하고, 상기 승산기 1026은 상기 입력된 심불열 R4의 각각의 심불들과 상기 입력되어진 입력신호 a3을 승산하여 길이 24인 심불열을 상기 배타적 가산기 1040으로 출력한다. 상기 승산기 1028은 상기 입력된 심불열 R5의 각각의 심불들과 상기 입력되어진 입력신호 a4를 승산하여 길이 24인 심불열을 상기 배타적 가산기 1040으로 출력하고, 상기 승산기 1030은 상기 입력된 심불열 R6의 각각의 심불들과 상기 입력되어진 입력신호 a5를 승산하여 길이 24인 심불열을 상기 배타적 가산기 1040으로 출력한다. 상기 승산기 1032는 상기 입력된 심불열 R7의 각각의 심불들과 상기 입력되어진 입력신호 a6을 승산하여 길이 24인 심불열을 상기 배타적 가산기 1040으로 출력한다. 그러면, 상기 배타적 가산기 1040은 상기 입력된 7개의 길이 24인 심불열들을 심볼 단위로 배타적 가산하여 길이 24인 부호화 심불열을 출력한다.

전술한, (24,1)부호기, (24,2)부호기, (24,3)부호기, (24,4)부호기, (24,5)부호기, (24,6)부호기, (24,7)부호기는 부호기 구조 관점에서 직교부호의 구조로부터 유추할 수 있다는 공통점을 가지고 있다. 즉, (24,1)부호기에서는 사용되었던 (2,1) 직교부호를 사용한 것이고, (24,2)부호기에서는 사용되었던 (4,2) 직교부호를 사용한 것이고, (24,3)부호기에서는 사용되었던 (8,3) 직교부호를 사용한 것이고, (24,4)부호기에서는 사용되었던 (16,4) 직교부호를 사용한 것이고, (24,5)부호기에서는 사용되었던 (32,5) 직교부호를 사용한 것이고, (24,6)부호기에서 사용되어졌던 (32,6) 확장된 직교부호는 (32,5)부호기에서 마스크함수라는 추가적인 1개의 부호어의 기저(basis)를 사용하여 확장한 것이고, (24,7)부호기에서 사용되어졌던 (32,7) 확장된 직교부호는 (32,5)부호기에서 마스크함수라는 추가적인 2개의 부호어의 기저(basis)를 사용하여 확장하였다. 따라서, 상기 (24,1)부호기, (24,2)부호기, (24,3)부호기, (24,4)부호기, (24,5)부호기, (24,6)부호기, (24,7)부호기는 상기와 같은 공통점을 가지고 있고, 이를 이용하여 하기에 나타날 제1 실시 예에서는 상기 서로 다른 크기를 가지는 (24,1)부호기, (24,2)부호기, (24,3)부호기, (24,4)부호기, (24,5)부호기, (24,6)부호기, (24,7)부호기를 하나의 부호기로 부호화 할 수 있는 부호기를 나타낼 것이다.

제1실시 예(부호기)

도 5는 상기에서 설명한 바와 같이 서로 다른 길이인 (24,1)부호기, (24,2)부호기, (24,3)부호기, (24,4)부호기, (24,5)부호기, (24,6)부호기, (24,7)부호기를 모두 수행할 수 있는 부호기의 구조를 나타낸다. 즉, 상기 도 5에서는 1~7비트의 입력 정보비트들을 입력하고, 상기 1~7비트의 입력 정보비트들 각각을 길이 2, 4, 8, 16 또는 32의 서로 다른 월시부호들 또는 마스크들로 부호화하여 24개의 부호화 심볼들로 이루어진 부호화 심불열로 출력하는 이동통신시스템에서의 부호기의 구성을 보이고 있다.

상기 도 5를 참조하여 본 발명의 실시 예에 따른 부호기의 구성을 살펴보면, 제어기 510은 상기 입력된 정보비트들의 비트 수를 검사하여 상기 부호화를 위한 제어를 수행한다. 즉, 상기 제어기 510은 상기 입력 정보비트들로 1비트가 입력되면 월시부호 발생기 500과 마스크 발생기 505가 길이 2를 가지는 1개의 월시부호를 출력하도록 제어한다. 또한,

반복기 550의 반복 횟수를 32회로 결정하며, 천공기 560으로는 상기 1비트의 입력 정보비트들에 대응한 40개의 천공 위치들을 천공하도록 제어한다. 한편, 상기 제어기 510은 상기 입력 정보비트들로 2비트가 입력되면 월시부호 발생기 500과 마스크 발생기 505가 길이 4를 가지는 서로 다른 2개의 월시부호들을 출력하도록 제어한다. 또한, 반복기 550의 반복 횟수를 8회로 결정하며, 천공기 560으로는 상기 2비트의 입력 정보비트들에 대응한 8개의 천공 위치들을 천공하도록 제어한다. 한편, 상기 제어기 510은 상기 입력 정보비트들로 3비트가 입력되면 월시부호 발생기 500과 마스크 발생기 505가 길이 8을 가지는 서로 다른 3개의 월시부호들을 출력하도록 제어한다. 또한, 상기 반복기 550의 반복 횟수를 4회로 결정하며, 상기 천공기 560으로는 상기 3비트의 입력 정보비트들에 대응한 8개의 천공 위치들을 천공하도록 제어한다. 한편, 상기 제어기 510은 상기 입력 정보비트들로 4비트가 입력되면 월시부호 발생기 500과 마스크 발생기 505가 길이 16을 가지는 서로 다른 4개의 월시부호들을 출력하도록 제어한다. 또한, 상기 반복기 550의 반복 횟수를 2회로 결정하며, 상기 천공기 560으로는 상기 4비트의 입력 정보비트들에 대응한 8개의 천공 위치들을 천공하도록 제어한다. 한편, 상기 제어기 510은 상기 입력 정보비트들로 5비트가 입력되면 월시부호 발생기 500과 마스크 발생기 505가 길이 32를 가지는 서로 다른 5개의 월시부호들을 출력하도록 제어한다. 또한, 상기 반복기 550의 반복 횟수를 1회로 결정하며, 상기 천공기 560으로는 상기 5비트의 입력 정보비트들에 대응한 8개의 천공 위치들을 천공하도록 제어한다. 한편, 상기 제어기 510은 상기 입력 정보비트들로 6비트가 입력되면 월시부호 발생기 500과 마스크 발생기 505가 길이 32를 가지는 서로 다른 6개의 월시부호들과 마스크들을 출력하도록 제어한다. 또한, 상기 반복기 550의 반복 횟수를 1회로 결정하며, 상기 천공기 560으로는 상기 6비트의 입력 정보비트들에 대응한 8개의 천공 위치들을 천공하도록 제어한다. 한편, 상기 제어기 510은 상기 입력 정보비트들로 7비트가 입력되면 월시부호 발생기 500과 마스크 발생기 505가 길이 32를 가지는 서로 다른 7개의 월시부호들과 마스크들을 출력하도록 제어한다. 또한, 상기 반복기 550의 반복 횟수를 1회로 결정하며, 상기 천공기 560으로는 상기 7비트의 입력 정보비트들에 대응한 8개의 천공 위치들을 천공하도록 제어한다.

상기 월시부호 발생기 500은 상기 제어기 510으로부터의 제어에 의해 길이 2,4,8,16,32의 월시부호들을 선택적으로 출력한다. 예컨대, 1비트의 입력 정보비트들이 입력되면 상기 제어기 510의 제어에 의해 상기 월시부호 발생기 500은 길이 2의 1개의 월시부호들을 출력한다. 또한, 2비트의 입력 정보비트들이 입력되면 상기 제어기 510의 제어에 의해 상기 월시부호 발생기 500은 길이 4의 서로 다른 2개의 월시부호들을 출력한다. 또한, 3비트의 입력 정보비트들이 입력되면 상기 제어기 510의 제어에 의해 상기 월시부호 발생기 500은 길이 8의 서로 다른 3개의 월시부호들을 출력한다. 또한, 4비트의 입력 정보비트들이 입력되면 상기 제어기 510의 제어에 의해 상기 월시부호 발생기 500은 길이 16의 서로 다른 4개의 월시부호들을 출력한다. 또한, 5비트의 입력 정보비트들이 입력되면 상기 제어기 510의 제어에 의해 상기 월시부호 발생기 500은 길이 32의 서로 다른 5개의 월시부호들을 출력한다. 또한, 6비트의 입력 정보비트들이 입력되면 상기 제어기 510의 제어에 의해 상기 월시부호 발생기 500은 길이 32의 서로 다른 5개의 월시부호들을 출력한다. 또한, 7비트의 입력 정보비트들이 입력되면 상기 제어기 510의 제어에 의해 상기 월시부호 발생기 500은 길이 32의 서로 다른 5개의 월시부호들을 출력한다.

상기 마스크 발생기 505는 상기 제어기 510으로부터의 제어에 의해 길이 32의 마스크들을 선택적으로 출력한다. 예컨대, 1~5비트의 입력 정보비트들이 입력되면 상기 제어기 510의 제어에 의해 상기 마스크 발생기 505는 마스크들을 출력하지 않는다. 하지만, 6~7비트의 입력 정보비트들이 입력되면 상기 제어기 510의 제어에 의해 상기 마스크 발생기 505는 길이 32의 서로 다른 2개의 마스크들을 출력한다. 한편, 도면에서도 보여지고 있는 바와 같이 상기 마스크 발생기 505는 상기 제어기 510으로부터의 제어를 받지 않고 2개의 서로 다른 마스크들을 연속하여 출력하도록 구현될 수 있다.

승산기들 520 내지 532는 상기 1~7비트의 입력 정보비트들과 상기 월시부호 발생기 500과 상기 마스크 발생기 505로부터의 월시부호들 및 마스크들을 일대일로 승산하여 2,4,8,16 또는 32개의 부호화 심볼들로 이루어진 부호화 심볼 열들을 출력한다. 배타적 가산기 540은 상기 승산기들 520 내지 532로부터의 부호화 심볼열들을 배타적 가산하여 2, 4,8,16 또는 32개의 부호화 심볼들로 이루어지는 하나의 부호화 심볼열을 출력한다. 상기 반복기 550은 1비트의 입력

정보가 입력되면 상기 제어기 510으로부터의 제어에 의해 상기 배타적 가산기 540으로부터의 상기 부호화 심볼열을 소정 횟수 반복하여 64개의 부호화 심볼들로 이루어진 부호화 심볼열을 출력한다. 반면, 상기 반복기 550은 2~7비트의 입력정보가 입력되면 상기 제어기 510으로부터의 제어에 의해 상기 배타적 가산기 540으로부터의 상기 부호화 심볼열을 1,2,4 또는 8번 반복하여 32개의 부호화 심볼들로 이루어진 부호화 심볼열을 출력한다. 메모리 670은 상기 1비트의 입력 정보비트들에 대응한 40개의 천공 위치들, 상기 2비트의 입력 정보비트들에 대응한 8개의 천공 위치들, 상기 3비트의 입력 정보비트들에 대응한 8개의 천공 위치들, 상기 4비트의 입력 정보비트들에 대응한 8개의 천공 위치들, 상기 5비트의 입력 정보비트들에 대응한 8개의 천공 위치들, 상기 6비트의 입력 정보비트들에 대응한 8개의 천공 위치들과 상기 7비트의 입력 정보비트들에 대응한 8개의 천공 위치들을 저장한다. 상기 1비트의 입력 정보비트들에 대응하여 상기 메모리 570에 저장된 40개의 천공 위치들은 모든 짹수 번째와 1,3,5,7,9,11,13,15번째 심볼들이며, 2비트의 입력 정보비트들에 대응하여 상기 메모리 570에 저장된 8개의 천공 위치들은 0,4,8,12,16,20,24,28번째 심볼들이며, 3비트의 입력 정보비트들에 대응하여 상기 메모리 570에 저장된 8개의 천공 위치들은 0,3,5,6,7,8,16,24번째 심볼들이며, 4비트의 입력 정보비트들에 대응하여 상기 메모리 570에 저장된 8개의 천공 위치들은 0,1,2,3,4,5,6,16번째 심볼들이며, 5비트의 입력 정보비트들에 대응하여 상기 메모리 570에 저장된 8개의 천공 위치들은 0,1,2,3,4,5,6,7번째 심볼들이며, 6비트의 입력 정보비트들에 대응하여 상기 메모리 570에 저장된 8개의 천공 위치들은 0,1,2,3,4,5,6,7번째 심볼들이며, 상기 7비트의 입력 정보비트들에 대응하여 상기 메모리 570에 저장된 8개의 천공 위치들은 0,1,2,3,4,5,6,7번째 심볼들이다.

상기 입력 정보비트들의 비트 수에 대응한 천공 위치들을 저장하는 메모리의 메모리 테이블을 표로서 보이면 하기 < 표 1> 과 같다.

[표 1]

입력 정보비트 수	천 공 위 치
1	짜수 번째 심볼들 및 1,3,5,7,9,11,13,15번째 심볼
2	0,4,8,12,16,20,24,28번째 심볼
3	0,3,5,6,7,8,16,24번째 심볼
4	0,1,2,3,4,5,6,16번째 심볼
5	0,1,2,3,4,5,6,7번째 심볼(또는 0,4,8,12,16,20,24,28번째 심볼)
6	0,1,2,3,4,5,6,7번째 심볼(또는 0,4,8,12,16,20,24,28번째 심볼)
7	0,1,2,3,4,5,6,7번째 심볼(또는 0,4,8,12,16,20,24,28번째 심볼)

상기 천공기 560은 입력정보가 1비트이면 상기 반복기 550으로부터의 상기 64개 부호화 심볼들로 이루어진 부호화 심볼열을 입력하고, 상기 제어기 510으로부터의 제어에 의해 상기 64개의 부호화 심볼들 중 상기 메모리 570으로부터 독출한 40개의 해당 천공 위치들의 부호화 심볼들을 천공하여 상기 24개의 부호화 심볼들로 이루어진 상기 부호화 심볼열을 출력한다. 또한 상기 천공기 560은 입력정보가 2~7비트이면 상기 반복기 550으로부터의 상기 32개 부호화 심볼들로 이루어진 부호화 심볼열을 입력하고, 상기 제어기 510으로부터의 제어에 의해 상기 32개의 부호화 심볼들 중 상기 메모리 570으로부터 독출한 8개의 해당 천공 위치들의 부호화 심볼들을 천공하여 상기 24개의 부호화 심볼들로 이루어진 상기 부호화 심볼열을 출력한다. 즉, 상기 천공기 560은 1비트의 입력 정보비트들이 입력되면 상기 제어기 510의 제어에 의해 상기 반복기 550으로부터 출력되는 64개의 부호화 심볼들 중 모든 짹수 번째와 1,3,5,7,9,11,13,15번째 심볼들을 천공하여 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 출력한다. 또한, 상기 천공기 560은 2비트의 입력 정보비트들이 입력되면 상기 제어기 510의 제어에 의해 상기 반복기 550으로부터 출력되는 32개의 부호화 심볼들 중 0,4,8,12,16,20,24,28번째 심볼들을 천공하여 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 출력한다. 또한, 상기 천공기 560은 3비트의 입력 정보비트들이 입력되면 상기 제어기 510의 제어에 의해 상기 반복기 550으로부터 출력되는 32개의 부호화 심볼들 중 0,3,5,6,7,8,16,24번째 심볼들을 천공하여 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 출력한다. 또한, 상기 천공기 560은 4비트의 입력 정보비트들이 입력되면 상기 제어기 510의 제어에 의해 상기 반복기 550으로부터 출력되는 32개의 부호화 심볼들 중 0,1,2,3,4,5,6,16번째 심볼들을 천공하여 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 출력한다. 또한, 상기 천공기 560은 5비트의 입력 정보비트들이 입력되면 상기 제어기 510의 제어에 의해 상기 반복기 550으로부터 출력되는 32개의 부호화 심볼들 중 0,4,8,12,16,20,24,28번째 심볼들을 천공하여 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 출력한다.

번짜 심볼들 또는 0,1,2,3,4,5,6,7번짜 심볼들을 천공하여 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 출력한다. 또한, 상기 천공기 560은 7비트의 입력 정보비트들이 입력되면 상기 제어기 510의 제어에 의해 상기 반복기 550으로부터 출력되는 32개의 부호화 심볼들 중 0,4,8,12,16,20,24,28번짜 심볼들 또는 0,1,2,3,4,5,6,7번짜 심볼들을 천공하여 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 출력한다. 또한, 상기 천공기 560은 7비트의 입력 정보비트들이 입력되면 상기 제어기 510의 제어에 의해 상기 반복기 550으로부터 출력되는 32개의 부호화 심볼들 중 0,4,8,12,16,20,24,28번짜 심볼들 또는 0,1,2,3,4,5,6,7번짜 심볼들을 천공하여 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 출력한다. 또한, 상기 천공기 560은 7비트의 입력 정보비트들이 입력되면 상기 제어기 510의 제어에 의해 상기 반복기 550으로부터 출력되는 32개의 부호화 심볼들 중 0,4,8,12,16,20,24,28번짜 심볼들 또는 0,1,2,3,4,5,6,7번짜 심볼들을 천공하여 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 출력한다.

먼저 상술한 도 5를 참조하여 상기 부호기를 설명하기 위해서, (24,1) 부호기로 동작할 때, (24,2) 부호기로 동작할 때, (24,3) 부호기로 동작할 때, (24,4) 부호기로 동작할 때, (24,5) 부호기로 동작할 때, (24,6) 부호기로 동작할 때 및 (24,7) 부호기로 동작할 때를 구분하여 설명하기로 한다.

첫 번째로, (24,7) 부호기로의 동작을 살펴보면, 7비트의 데이터를 지시자 a0,a1,a2,a3,a4,a5,a6이 상기 부호기에 입력이 되면, 상기 7비트의 데이터를 지시자가 상기 부호기로 입력되어진다는 비트 정보가 제어기 510으로 입력되어진다. 그러면, 상기 제어기 510은 월시부호 발생기 500으로 길이 32를 가지는 월시부호를 생성하도록 하는 제어신호를 출력한다. 그러면 상기 월시부호 발생기 500은 길이 32인 월시부호 W1, W2, W4, W8, W16을 출력한다. 상기 월시부호 발생기 500으로부터 발생된 상기 W1, W2, W4, W8, W16은 각각에 대응하는 승산기로 제공된다. 예컨대, 상기 W1은 승산기 520으로 제공되며, 상기 W2는 승산기 522로 제공된다. 상기 W4는 승산기 524로 제공되며, 상기 W8은 승산기 526으로 제공된다. 마지막으로 상기 W16은 승산기 528로 제공된다. 상기 월시부호 발생기 500으로부터 발생되는 길이 32의 월시부호는 하기 < 표 2> 과 같다.

[표 2]

월시번호	월시부호
W1	0101 0101 0101 0101 0101 0101 0101 0101 0101 0101
W2	0011 0011 0011 0011 0011 0011 0011 0011 0011 0011
W4	0000 1111 0000 1111 0000 1111 0000 1111
W8	0000 0000 1111 1111 0000 0000 1111 1111
W16	0000 0000 0000 0000 1111 1111 1111 1111

이와 동시에, 마스크 발생기 505는 마스크 M1 = 0111 0111 0010 0100 0110 0000 0000 0000을 발생시켜 승산기 530으로 출력하며, 마스크 M2 = 0010 0110 0101 0100 0101 0100 0000을 발생시켜 승산기 532로 출력한다.

한편, 상기 7개의 입력정보 비트들은 상기 승산기들 중 대응하는 각각의 상기 승산기로 제공된다. 상기 7개의 입력정보 비트들과 상기 승산기들의 대응 관계는 하기 < 표 3> 에서 보이고 있는 바와 같다.

[표 3]

입력정보 비트	승산기
a0	승산기 520
a1	승산기 522
a2	승산기 524
a3	승산기 526
a4	승산기 528
a5	승산기 530
a6	승산기 532

따라서, 상기 승산기들 각각으로 제공되는 입력정보 비트와 월시부호 및 마스크는 하기 < 표 4> 과 같이 나타낼 수 있다.

[표 4]

승산기	월시부호/마스크	입력정보 비트
승산기 520	W1 = 0101 0101 0101 0101 0101 0101 0101 0101	a0
승산기 522	W2 = 0011 0011 0011 0011 0011 0011 0011 0011	a1
승산기 524	W4 = 0000 1111 0000 1111 0000 1111 0000 1111	a2
승산기 526	W8 = 0000 0000 1111 1111 0000 0000 1111 1111	a3
승산기 528	W16 = 0000 0000 0000 0000 1111 1111 1111 1111	a4
승산기 530	M1 = 0111 0111 0010 0100 0110 0000 0000 0000	a5
승산기 532	M2 = 0010 0110 0101 0100 0101 0100 0100 0000	a6

상기 승산기들은 상기 < 표 4> 에서 보이고 있는 바와 같이 자신에게로 제공되는 입력정보 비트를 월시부호/마스크를 구성하는 각각의 심볼들과 승산하여 가산기 540으로 출력한다.

즉, 상기 승산기 520은 a0을 W1 각각의 심볼마다 승산하여 가산기 540으로 출력하고, 상기 승산기 522는 a1을 W2 각각의 심볼마다 승산하여 상기 가산기 540으로 출력하며, 상기 승산기 524는 a2를 W4 각각의 심볼마다 승산하여 상기 가산기 540으로 출력한다. 한편, 상기 승산기 526은 a3을 W8 각각의 심볼마다 승산하여 상기 가산기 540으로 출력하고, 상기 승산기 528은 a4를 W16 각각의 심볼마다 승산하여 상기 가산기 540으로 출력한다. 또한, 상기 승산기 530은 a5를 M1 각각의 심볼마다 승산하여 상기 가산기 540으로 출력하고, 상기 승산기 532는 a6을 M2 각각의 심볼마다 승산하여 상기 가산기 540으로 출력한다. 상기 가산기 540은 배타적 가산기를 사용한다.

그러면, 상기 배타적 가산기 540은 상기 승산기 520, 522, 524, 526, 528, 530, 532로부터 출력되어진 길이 32인 모든 심볼 열들을 심볼 단위로 배타적 가산하여 반복기 550으로 출력한다.

상기 배타적 가산기 540으로부터 출력되는 길이 32의 부호화 심볼열(Ws)은 하기 < 수학식 8> 로 표현될 수 있다.

수학식 8

$$Ws = (W1 \times a0) + (W2 \times a1) + (W4 \times a2) + (W8 \times a3) + (W16 \times a4) + (M1 \times a5) + (M2 \times a6)$$

이때, 상기 제어기 510은 상기 반복기 550이 상기 배타적 가산기 540의 출력을 반복할 횟수를 제어하는 제어신호를 출력한다. (24,7) 부호기에서는 상기 배타적 가산기 540으로부터 출력되는 심볼 열이 길이 32를 가짐에 따라 상기 반복기 550의 반복 횟수는 1회가 될 것이다. 따라서, 상기 제어기 510은 상기 반복기 550으로 입력신호 1회 반복(즉, 입력신호를 그대로 출력) 하라는 제어신호를 출력한다. 이대 응답하여 상기 반복기 550은 상기 배타적 가산기 540으로부터 출력되어진 길이 32인 심볼들을 그대로 천공기 560으로 출력한다.

이 시점에서 상기 제어기 510은 천공기 560으로 (24,7) 부호에 대한 천공 위치를 천공하라는 제어신호를 출력한다. 그러면, 상기 천공기 560은 천공위치가 저장되어 있는 메모리 570으로부터 천공위치를 입력받아 상기 반복기 550으로부터 제공되는 길이 32의 부호화 심볼들 중 8개의 해당 부호화 심볼들을 천공한다. 즉, 상기 천공기 560은 상기 반복기 550으로부터 제공되는 길이 32의 부호화 심볼들 중 상기 메모리 570으로부터의 천공위치에 해당하는 0,4,8,12,16,20,24,28번째 심볼들을 천공한다.

하지만, 다른 예로서 하드웨어 복잡도를 줄이기 위해 0,1,2,3,4,5,6,7번째 심볼들을 천공위치로 하여 길이 32의 부호화 심볼들을 천공할 수 있다. 이 경우 상기 마스크 발생기 505는 $M1 = 0000\ 0000\ 1110\ 1000\ 1101\ 1000\ 1100\ 00$ 00과 $M2 = 0000\ 0000\ 1100\ 0000\ 0111\ 1110\ 0010\ 1000$ 을 마스크로서 발생한다. 즉, 상기 메모리 570으로부터의 천공위치가 0,1,2,3,4,5,6,7인 경우 상기 천공기 560은 상기 길이 32의 부호화 심볼들 중 0,1,2,3,4,5,6,7번째 부호화 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

두 번째로, (24,6) 부호기로 등작할 때를 살펴보면, 6비트의 데이터를 지시자 $a_0, a_1, a_2, a_3, a_4, a_5$ 이 상기 부호기에 입력이 되고, 나머지 데이터를 지시자 a_6 은 초기화되어 입력된다. 상기 초기화에 의해 상기 a_6 으로는 0이 입력된다. 한편, 상기 제어기 510으로는 6비트의 데이터를 지시자가 상기 부호기로 입력되어진다는 비트 정보가 입력되어진다. 그러면, 상기 제어기 510은 월시부호 발생기 500으로 길이 32을 가지는 월시부호를 생성하도록 하는 제어신호를 출력한다. 그러면 상기 월시부호 발생기 500은 길이 32인 월시부호 $W_1, W_2, W_4, W_8, W_{16}$ 을 출력한다. 상기 월시부호 발생기 500으로부터 발생된 상기 $W_1, W_2, W_4, W_8, W_{16}$ 은 각각에 대응하는 승산기로 제공된다. 예컨대, 상기 W_1 은 승산기 520으로 제공되며, 상기 W_2 는 승산기 522로 제공된다. 상기 W_4 는 승산기 524로 제공되며, 상기 W_8 은 승산기 526으로 제공된다. 상기 W_{16} 은 승산기 528로 제공된다. 상기 월시부호 발생기 500으로부터 발생되는 길이 32의 월시부호는 하기 < 표 5> 과 같다.

[표 5]

월시번호	월시부호
W_1	0101 0101 0101 0101 0101 0101 0101 0101
W_2	0011 0011 0011 0011 0011 0011 0011 0011
W_4	0000 1111 0000 1111 0000 1111 0000 1111
W_8	0000 0000 1111 1111 0000 0000 1111 1111
W_{16}	0000 0000 0000 0000 1111 1111 1111 1111

이와 동시에, 마스크 발생기 505는 마스크 $M1 = 0000\ 0000\ 1110\ 1000\ 1101\ 1000\ 1100\ 0000$ 을 발생시켜 승산기 530으로 출력한다.

한편, 상기 6개의 입력정보 비트들은 상기 승산기들 중 대응하는 각각의 상기 승산기로 제공된다. 상기 6개의 입력정보 비트들과 상기 승산기들의 대응 관계는 하기 < 표 6> 에서 보이고 있는 바와 같다.

[표 6]

입력정보 비트	승산기
a0	승산기 520
a1	승산기 522
a2	승산기 524
a3	승산기 526
a4	승산기 528
a5	승산기 530

따라서, 상기 승산기들 각각으로 제공되는 입력정보 비트와 월시부호들은 하기 < 표 7> 과 같이 나타낼 수 있다.

[표 7]

승산기	월시부호/마스크	입력정보 비트
승산기 520	W1 = 0101 0101 0101 0101 0101 0101 0101 0101	a0
승산기 522	W2 = 0011 0011 0011 0011 0011 0011 0011 0011	a1
승산기 524	W4 = 0000 1111 0000 1111 0000 1111 0000 1111	a2
승산기 526	W8 = 0000 0000 1111 1111 0000 0000 1111 1111	a3
승산기 528	W16 = 0000 0000 0000 0000 1111 1111 1111 1111	a4
승산기 530	M1 = 0000 0000 1110 1000 1101 1000 1100 0000	a5

상기 승산기들은 상기 < 표 7> 에서 보이고 있는 바와 같이 자신에게로 제공되는 입력정보 비트를 월시부호를 구성하는 각각의 심볼들과 승산하여 가산기 540으로 출력한다.

즉, 상기 승산기 520은 a0을 W1 각각의 심볼마다 승산하여 가산기 540으로 출력하고, 상기 승산기 522는 a1을 W2 각각의 심볼마다 승산하여 상기 가산기 540으로 출력한다. 한편, 상기 승산기 524는 a2를 W4 각각의 심볼마다 승산하여 상기 가산기 540으로 출력하며, 상기 승산기 526은 a3을 W8 각각의 심볼마다 승산하여 상기 가산기 540으로 출력한다. 상기 가산기 540은 배타적 가산기를 사용한다.

한편, 승산기 532로는 상기 초기화로 인해 0의 값을 가지는 a6이 각각 인가됨에 따라 상기 마스크 발생기 505로부터의 M2에 관계없이 상기 승산기 532의 출력은 상기 가산기 540의 출력에 영향을 미치지 못한다. 즉, 상기 마스크 생성기 505로부터 승산기 532로 입력되어지는 심볼열들이 어떤 값이든 상관없이 상기 승산기 532로부터는 0인 심볼열들이 출력된다. 따라서, 상기 승산기 532의 출력은 상기 배타적 가산기 540에 입력되어도 아무런 영향을 미치지 않는다. 상기와 같이 a6을 0으로 초기화하는 것은 상기 승산기 532의 출력을 차단하는 스위치 동작과 유사하다.

그러면, 상기 배타적 가산기 540은 상기 승산기 520, 522, 524, 526, 528, 530, 532로부터 출력되어진 길이 32인 모든 심볼 열들을 심볼 단위로 배타적 가산하여 반복기 550으로 출력한다.

상기 배타적 가산기 540으로부터 출력되는 길이 32의 부호화 심볼열(Ws)은 하기 < 수학식 9> 로 표현될 수 있다.

수학식 9

$$Ws = (W1 \times 0) + (W2 \times 1) + (W4 \times 2) + (W8 \times 3) + (W16 \times 4) + (M1 \times 5)$$

이때, 상기 제어기 510은 상기 반복기 550이 상기 배타적 가산기 540의 출력을 반복할 횟수를 제어하는 제어신호를 출력한다. (24,6) 부호기에서는 상기 배타적 가산기 540으로부터 출력되는 심볼 열이 길이 32를 가짐에 따라 상기 반복기 550의 반복 횟수는 1회가 될 것이다. 따라서, 상기 제어기 510은 상기 반복기 550으로 입력신호 1회 반복(즉, 입력신호를 그대로 출력) 하라는 제어신호를 출력한다. 이에 응답하여 상기 반복기 550은 상기 배타적 가산기 540으로부

터 출력되어진 길이 32인 심볼들을 그대로 천공기 560으로 출력한다.

이 시점에서 상기 제어기 510은 천공기 560으로 (24,6)부호에 대한 천공 위치를 천공하라는 제어신호를 출력한다. 그러면, 상기 천공기 560은 천공위치가 저장되어 있는 메모리 570으로부터 천공위치를 입력받아 상기 반복기 550으로부터 제공되는 길이 32의 부호화 심볼들 중 8개의 해당 부호화 심볼들을 천공한다. 즉, 상기 천공기 560은 상기 반복기 550으로부터 제공되는 길이 32의 부호화 심볼들 중 상기 메모리 570으로부터의 천공위치에 해당하는 0,4,8,12,16,2,0,24,28번째 심볼들을 천공한다.

하지만, 다른 예로서 하드웨어 복잡도를 줄이기 위해 0,1,2,3,4,5,6,7번째 심볼들을 천공위치로 하여 길이 32의 부호화 심볼들을 천공할 수 있다. 이 경우 상기 마스크 발생기 505는 M1=0000 0000 1110 1000 1101 1000 1100 00 00을 마스크로서 발생한다. 즉, 상기 메모리 570으로부터의 천공위치가 0,1,2,3,4,5,6,7인 경우 상기 천공기 560은 상기 길이 32의 부호화 심볼들 중 0,1,2,3,4,5,6,7번째 부호화 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

세 번째로, (24,5)부호기로 동작할 때를 살펴보면, 5비트의 데이터를 지시자 a0,a1,a2,a3,a4이 상기 부호기에 입력이 되고, 나머지 데이터를 지시자 a5,a6은 초기화되어 입력된다. 상기 초기화에 의해 상기 a5,a6으로는 0이 입력된다. 한편, 상기 제어기 510으로는 5비트의 데이터를 지시자가 상기 부호기로 입력되어진다는 비트 정보가 입력되어진다. 그러면, 상기 제어기 510은 월시부호 발생기 500으로 길이 32을 가지는 월시부호를 생성하도록 하는 제어신호를 출력한다. 그러면 상기 월시부호 발생기 500은 길이 32인 월시부호 W1, W2, W4, W8, W16을 출력한다. 상기 월시부호 발생기 500으로부터 발생된 상기 W1, W2, W4, W8, W16은 각각에 대응하는 승산기로 제공된다. 예컨대, 상기 W1은 승산기 520으로 제공되며, 상기 W2는 승산기 522로 제공된다. 상기 W4는 승산기 524로 제공되며, 상기 W8은 승산기 526으로 제공된다. 상기 W16은 승산기 528로 제공된다. 상기 월시부호 발생기 500으로부터 발생되는 길이 32의 월시부호는 하기 < 표 8> 과 같다.

[표 8]

월시번호	월시부호
W1	0101 0101 0101 0101 0101 0101 0101 0101 0101
W2	0011 0011 0011 0011 0011 0011 0011 0011 0011
W4	0000 1111 0000 1111 0000 1111 0000 1111
W8	0000 0000 1111 1111 0000 0000 1111 1111
W16	0000 0000 0000 0000 1111 1111 1111 1111

한편, 상기 5개의 입력정보 비트들은 상기 승산기들 중 대응하는 각각의 상기 승산기로 제공된다. 상기 5개의 입력정보 비트들과 상기 승산기들의 대응 관계는 하기 < 표 9> 에서 보이고 있는 바와 같다.

[표 9]

입력정보 비트	승산기
a0	승산기 520
a1	승산기 522
a2	승산기 524
a3	승산기 526
a4	승산기 528

따라서, 상기 승산기들 각각으로 제공되는 입력정보 비트와 월시부호들은 하기 < 표 10> 과 같이 나타낼 수 있다.

[표 10]

승산기	월시부호/마스크	입력정보 비트
승산기 520	W1 = 0101 0101 0101 0101 0101 0101 0101 0101	a0
승산기 522	W2 = 0011 0011 0011 0011 0011 0011 0011 0011	a1
승산기 524	W4 = 0000 1111 0000 1111 0000 1111 0000 1111	a2
승산기 526	W8 = 0000 0000 1111 1111 0000 0000 1111 1111	a3
승산기 528	W16 = 0000 0000 0000 1111 1111 1111 1111	a4

상기 승산기들은 상기 < 표 10> 에서 보이고 있는 바와 같이 자신에게로 제공되는 입력정보 비트를 월시부호를 구성하는 각각의 심볼들과 승산하여 가산기 540으로 출력한다.

즉, 상기 승산기 520은 a0을 W1 각각의 심볼마다 승산하여 가산기 540으로 출력하고, 상기 승산기 522는 a1을 W2 각각의 심볼마다 승산하여 상기 가산기 540으로 출력한다. 한편, 상기 승산기 524는 a2를 W4 각각의 심볼마다 승산하여 상기 가산기 540으로 출력하며, 상기 승산기 526은 a3을 W8 각각의 심볼마다 승산하여 상기 가산기 540으로 출력한다. 상기 가산기 540은 배타적 가산기를 사용한다.

한편, 승산기 530, 532로는 상기 초기화로 인해 0의 값을 가지는 a5, a6이 각각 인가됨에 따라 상기 마스크 발생기 505로부터의 M1, M2에 관계없이 상기 승산기 530, 532의 출력은 상기 가산기 540의 출력에 영향을 미치지 못한다. 즉, 상기 마스크 생성기 505로부터 승산기 530, 532로 입력되어지는 심볼열들이 어떤 값이든 상관없이 상기 승산기 532로부터는 0인 심볼열들이 출력된다. 따라서, 상기 승산기 532의 출력은 상기 배타적 가산기 540에 입력되어도 아무런 영향을 미치지 않는다. 상기와 같이 a5, a6을 0으로 초기화하는 것은 상기 승산기 530, 532의 출력을 차단하는 스위치 동작과 유사하다.

그러면, 상기 배타적 가산기 540은 상기 승산기 520, 522, 524, 526, 528, 530, 532로부터 출력되어진 길이 32인 모든 심볼 열들을 심볼 단위로 배타적 가산하여 반복기 550으로 출력한다.

상기 배타적 가산기 540으로부터 출력되는 길이 32의 부호화 심볼열(Ws)은 하기 < 수학식 10> 로 표현될 수 있다.

수학식 10

$$Ws = (W1 \times 0) + (W2 \times 1) + (W4 \times 2) + (W8 \times 3) + (W16 \times 4)$$

이때, 상기 제어기 510은 상기 반복기 550이 상기 배타적 가산기 540의 출력을 반복할 횟수를 제어하는 제어신호를 출력한다. (24,5) 부호기에서는 상기 배타적 가산기 540으로부터 출력되는 심볼 열이 길이 32를 가짐에 따라 상기 반복기 550의 반복 횟수는 1회가 될 것이다. 따라서, 상기 제어기 510은 상기 반복기 550으로 입력신호 1회 반복(즉, 입력신호를 그대로 출력) 하라는 제어신호를 출력한다. 이에 응답하여 상기 반복기 550은 상기 배타적 가산기 540으로부터 출력되어진 길이 32인 심볼들을 그대로 천공기 560으로 출력한다.

이 시점에서 상기 제어기 510은 천공기 560으로 (24,5) 부호에 대한 천공 위치를 천공하라는 제어신호를 출력한다. 그러면, 상기 천공기 560은 천공위치가 저장되어 있는 메모리 570으로부터 천공위치를 입력받아 상기 반복기 550으로부터 제공되는 길이 32의 부호화 심볼들 중 8개의 해당 부호화 심볼들을 천공한다. 즉, 상기 천공기 560은 상기 반복기 550으로부터 제공되는 길이 32의 부호화 심볼들 중 상기 메모리 570으로부터의 천공위치에 해당하는 0,4,8,12,16,2,0,24,28번째 심볼들을 천공한다.

하지만, 다른 예로서 하드웨어 복잡도를 줄이기 위해 0,1,2,3,4,5,6,7번째 심볼들을 천공위치로 하여 길이 32의 부호화 심볼들을 천공할 수 있다. 즉, 상기 메모리 570으로부터의 천공위치가 0,1,2,3,4,5,6,7인 경우 상기 천공기 560은 상기 길이 32의 부호화 심볼들 중 0,1,2,3,4,5,6,7번째 부호화 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

네 번째로, (24,4) 부호기로 동작할 때를 살펴보면, 4비트의 데이터를 지시자 a0,a1,a2,a3이 상기 부호기에 입력이 되고, 나머지 데이터를 지시자 a4,a5,a6은 초기화되어 입력된다. 상기 초기화에 의해 상기 a4,a5,a6으로는 0이 입력된다. 한편, 상기 제어기 510으로는 4비트의 데이터를 지시자가 상기 부호기로 입력되어진다는 비트 정보가 입력되어진다. 그러면, 상기 제어기 510은 월시부호 발생기 500으로 길이 16을 가지는 월시부호를 생성하도록 하는 제어신호를 출력한다. 그러면 상기 월시부호 발생기 500은 길이 16인 월시부호 W1, W2, W4, W8을 출력한다. 상기 월시부호 발생기 500으로부터 발생된 상기 W1, W2, W4, W8은 각각에 대응하는 승산기로 제공된다. 예컨대, 상기 W1은 승산기 520으로 제공되며, 상기 W2는 승산기 522로 제공된다. 상기 W4는 승산기 524로 제공되며, 상기 W8은 승산기 526으로 제공된다. 상기 월시부호 발생기 500으로부터 발생되는 길이 16의 월시부호는 하기 < 표 11> 과 같다.

[표 11]

월시번호	월시부호
W1	0101 0101 0101 0101
W2	0011 0011 0011 0011
W4	0000 1111 0000 1111
W8	0000 0000 1111 1111

한편, 상기 4개의 입력정보 비트들은 상기 승산기들 중 대응하는 각각의 상기 승산기로 제공된다. 상기 4개의 입력정보 비트들과 상기 승산기들의 대응 관계는 하기 < 표 12>에서 보이고 있는 바와 같다.

[표 12]

입력정보 비트	승산기
a0	승산기 520
a1	승산기 522
a2	승산기 524
a3	승산기 526

따라서, 상기 승산기들 각각으로 제공되는 입력정보 비트와 월시부호들은 하기 < 표 13>과 같이 나타낼 수 있다.

[표 13]

승산기	월시부호	입력정보 비트
승산기 520	W1 = 0101 0101 0101 0101	a0
승산기 522	W2 = 0011 0011 0011 0011	a1
승산기 524	W4 = 0000 1111 0000 1111	a2
승산기 526	W8 = 0000 0000 1111 1111	a3

상기 승산기들은 상기 < 표 13> 에서 보이고 있는 바와 같이 자신에게로 제공되는 입력정보 비트를 월시부호를 구성하는 각각의 심볼들과 승산하여 가산기 540으로 출력한다.

즉, 상기 승산기 520은 a0을 W1 각각의 심볼마다 승산하여 가산기 540으로 출력하고, 상기 승산기 522는 a1을 W2 각각의 심볼마다 승산하여 상기 가산기 540으로 출력한다. 한편, 상기 승산기 524는 a2를 W4 각각의 심볼마다 승산하여 상기 가산기 540으로 출력하며, 상기 승산기 526은 a3을 W8 각각의 심볼마다 승산하여 상기 가산기 540으로 출력한다. 상기 가산기 540은 배타적 가산기를 사용한다.

한편, 승산기 528, 530, 532로는 상기 초기화로 인해 0의 값을 가지는 a4, a5, a6이 각각 인가됨에 따라 상기 월시부호 발생기 500으로부터의 W16과 마스크 발생기 505로부터의 M1, M2에 관계없이 상기 승산기 528, 530, 532의 출력은 상기 가산기 540의 출력에 영향을 미치지 못한다. 즉, 상기 월시부호 생성기 500으로부터 승산기 528로 입력되어지는 W16 심볼열들이 어떤 값이든 상관없이 상기 승산기 528로부터는 0인 심볼열들이 출력된다. 또한, 상기 마스크 생성기 505로부터 승산기 530, 532로 입력되어지는 심볼열들이 어떤 값이든 상관없이 상기 승산기 530, 532로부터는 0인 심볼열들이 출력된다. 따라서, 상기 승산기 528, 530, 532의 출력은 상기 배타적 가산기 540에 입력되어도 아무런 영향을 미치지 않는다. 상기와 같이 a4,a5,a6을 0으로 초기화하는 것은 상기 승산기 528, 530, 532의 출력을 차단하는 스위치 동작과 유사하다.

그러면, 상기 배타적 가산기 540은 상기 승산기 520, 522, 524, 526, 528, 530, 532로부터 출력되어진 길이 16인 모든 심볼 열들을 심볼 단위로 배타적 가산하여 반복기 550으로 출력한다.

상기 배타적 가산기 540으로부터 출력되는 길이 32의 부호화 심볼열(Ws)은 하기 < 수학식 11> 로 표현될 수 있다.

수학식 11

$$Ws = (W1 \times 0) + (W2 \times 1) + (W4 \times 2) + (W8 \times 3)$$

이때, 상기 제어기 510은 상기 반복기 550이 상기 배타적 가산기 540의 출력을 반복할 횟수를 제어하는 제어신호를 출력한다. (24,4) 부호기에서는 상기 배타적 가산기 540으로부터 출력되는 심볼 열이 길이 16을 가짐에 따라 상기 반복기 550의 반복 횟수는 2회가 될 것이다. 따라서, 상기 제어기 510은 상기 반복기 550으로 입력신호 2회 반복하라는 제어신호를 출력한다. 이에 응답하여 상기 반복기 550은 상기 배타적 가산기 540으로부터 출력되어진 길이 16인 심볼들을 2회 반복한 길이 32의 천공기 560으로 출력한다.

이 시점에서 상기 제어기 510은 천공기 560으로 (24,4) 부호에 대한 천공 위치를 천공하라는 제어신호를 출력한다. 그러면, 상기 천공기 560은 천공위치가 저장되어 있는 메모리 570으로부터 천공위치를 입력받아 상기 반복기 550으로부터 제공되는 길이 32의 부호화 심볼들 중 8개의 해당 부호화 심볼들을 천공한다. 즉, 상기 천공기 560은 상기 반복기 550으로부터 제공되는 길이 32의 부호화 심볼들 중 상기 메모리 570으로부터의 천공위치에 해당하는 8 심볼을 천공한다. 예컨대, 상기 메모리 570으로부터의 천공위치가 0,1,2,3,4,5,6,16인 경우 상기 천공기 560은 상기 길이 32의 부호화 심볼들 중 0,1,2,3,4,5,6,16번째 부호화 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

다섯 번째로, (24,3) 부호기로 동작할 때를 살펴보면, 3비트의 데이터를 지시자 a0,a1,a2이 상기 부호기에 입력이 되고, 나머지 데이터를 지시자 a3,a4,a5,a6은 초기화되어 입력된다. 상기 초기화에 의해 상기 a3,a4,a5,a6으로는 0이 입력된다. 한편, 상기 제어기 510으로는 3비트의 데이터를 지시자가 상기 부호기로 입력되어진다는 비트 정보가 입력되어진다. 그러면, 상기 제어기 510은 월시부호 발생기 500으로 길이 8을 가지는 월시부호를 생성하도록 하는 제어신호를 출력한다. 그러면 상기 월시부호 발생기 500은 길이 8인 월시부호 W1, W2, W4을 출력한다. 상기 월시부호 발생기 500으로부터 발생된 상기 W1, W2, W4는 각각에 대응하는 승산기로 제공된다. 예컨대, 상기 W1은 승산기 520으로 제공되며, 상기 W2는 승산기 522로 제공된다. 상기 W4는 승산기 524로 제공된다. 상기 월시부호 발생기 500으로부터 발생되는 길이 8의 월시부호는 하기 < 표 14> 과 같다.

[표 14]

월시번호	월시부호
W1	0101 0101
W2	0011 0011
W4	0000 1111

한편, 상기 3개의 입력정보 비트들은 상기 승산기들 중 대응하는 각각의 상기 승산기로 제공된다. 상기 3개의 입력정보 비트들과 상기 승산기들의 대응 관계는 하기 < 표 15> 에서 보이고 있는 바와 같다.

[표 15]

입력정보 비트	승산기
a0	승산기 520
a1	승산기 522
a2	승산기 524

따라서, 상기 승산기들 각각으로 제공되는 입력정보 비트와 월시부호들은 하기 < 표 16> 과 같이 나타낼 수 있다.

[표 16]

승산기	월시부호	입력정보 비트
승산기 520	W1 = 0101 0101	a0
승산기 522	W2 = 0011 0011	a1
승산기 524	W4 = 0000 1111	a2

상기 승산기들은 상기 < 표 16> 에서 보이고 있는 바와 같이 자신에게로 제공되는 입력정보 비트를 월시부호를 구성하는 각각의 심볼들과 승산하여 가산기 540으로 출력한다.

즉, 상기 승산기 520은 a0을 W1 각각의 심볼마다 승산하여 가산기 540으로 출력하고, 상기 승산기 522는 a1을 W2 각각의 심볼마다 승산하여 상기 가산기 540으로 출력한다. 한편, 상기 승산기 524는 a2를 W4 각각의 심볼마다 승산하여 상기 가산기 540으로 출력한다. 상기 가산기 540은 배타적 가산기를 사용한다.

한편, 승산기 526, 528, 530, 532로는 상기 초기화로 인해 0의 값을 가지는 a3, a4, a5, a6이 각각 인가됨에 따라 상기 월시부호 발생기 500으로부터의 W8, W16과 마스크 발생기 505로부터의 M1, M2에 관계없이 상기 승산기 526, 528, 530, 532의 출력은 상기 가산기 540의 출력에 영향을 미치지 못한다. 즉, 상기 월시부호 생성기 500으로부터 승산기 526, 528로 입력되어지는 W8, W16 심볼열들이 어떤 값이든 상관없이 상기 승산기 526, 528로부터는 0인 심볼열들이 출력된다. 또한, 상기 마스크 생성기 505로부터 승산기 530, 532로 입력되어지는 심볼열들이 어떤 값이든 상

관없이 상기 승산기 530, 532로부터는 0인 심볼열들이 출력된다. 따라서, 상기 승산기 526, 528, 530, 532의 출력은 상기 배타적 가산기 540에 입력되어도 아무런 영향을 미치지 않는다. 상기와 같이 a3,a4,a5,a6을 0으로 초기화하는 것은 상기 승산기 526, 528, 530, 532의 출력을 차단하는 스위치 동작과 유사하다.

그러면, 상기 배타적 가산기 540은 상기 승산기 520, 522, 524, 526, 528, 530, 532로부터 출력되어진 길이 8인 모든 심볼 열들을 심볼 단위로 배타적 가산하여 반복기 550으로 출력한다.

상기 배타적 가산기 540으로부터 출력되는 길이 8의 부호화 심볼열(Ws)은 하기 < 수학식 12>로 표현될 수 있다.

수학식 12

$$Ws = (W1 \times 0) + (W2 \times 1) + (W4 \times 2)$$

이때, 상기 제어기 510은 상기 반복기 550이 상기 배타적 가산기 540의 출력을 반복할 횟수를 제어하는 제어신호를 출력한다. (24,3) 부호기에서는 상기 배타적 가산기 540으로부터 출력되는 심볼 열이 길이 8을 가짐에 따라 상기 반복기 550의 반복 횟수는 4회가 될 것이다. 따라서, 상기 제어기 510은 상기 반복기 550으로 입력신호 4회 반복(즉, 입력신호를 그대로 출력)하라는 제어신호를 출력한다. 이에 응답하여 상기 반복기 550은 상기 배타적 가산기 540으로부터 출력되어진 길이 8인 심볼들을 4회 반복한 길이 32의 천공기 560으로 출력한다.

이 시점에서 상기 제어기 510은 천공기 560으로 (24,3) 부호에 대한 천공 위치를 천공하라는 제어신호를 출력한다. 그러면, 상기 천공기 560은 천공위치가 저장되어 있는 메모리 570으로부터 천공위치를 입력받아 상기 반복기 550으로부터 제공되는 길이 32의 부호화 심볼들 중 8개의 해당 부호화 심볼들을 천공한다. 즉, 상기 천공기 560은 상기 반복기 550으로부터 제공되는 길이 32의 부호화 심볼들 중 상기 메모리 570으로부터의 천공위치에 해당하는 8 심볼을 천공한다. 예컨대, 상기 메모리 570으로부터의 천공위치가 0,3,5,6,7,8,16,24인 경우 상기 천공기 560은 상기 길이 32의 부호화 심볼들 중 0,3,5,6,7,8,16,24번째 부호화 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

여섯 번째로, (24,2) 부호기로 동작할 때를 살펴보면, 2비트의 데이터를 지시자 a0,a1이 상기 부호기에 입력이 되고, 나머지 데이터를 지시자 a2,a3,a4,a5,a6은 초기화되어 입력된다. 상기 초기화에 의해 상기 a2,a3,a4,a5,a6으로는 0이 입력된다. 한편, 상기 제어기 510으로는 2비트의 데이터를 지시자가 상기 부호기로 입력되어진다는 비트 정보가 입력되어진다. 그러면, 상기 제어기 510은 월시부호 발생기 500으로 길이 4를 가지는 월시부호를 생성하도록 하는 제어신호를 출력한다. 그러면 상기 월시부호 발생기 500은 길이 4인 월시부호 W1, W2를 출력한다. 상기 월시부호 발생기 500으로부터 발생된 상기 W1, W2는 각각에 대응하는 승산기로 제공된다. 예컨대, 상기 W1은 승산기 520으로 제공되며, 상기 W2는 승산기 522로 제공된다. 상기 월시부호 발생기 500으로부터 발생되는 길이 4의 월시부호는 하기 < 표 17>과 같다.

[표 17]

월시번호	월시부호
W1	0101
W2	0011

한편, 상기 2개의 입력정보 비트들은 상기 승산기들 중 대응하는 각각의 상기 승산기로 제공된다. 상기 2개의 입력정보 비트들과 상기 승산기들의 대응 관계는 하기 < 표 18> 에서 보이고 있는 바와 같다.

[표 18]

입력정보 비트	승산기
a0	승산기 520
a1	승산기 522

따라서, 상기 승산기들 각각으로 제공되는 입력정보 비트와 월시부호들은 하기 < 표 19> 와 같이 나타낼 수 있다.

[표 19]

승산기	월시부호	입력정보 비트
승산기 520	W1 = 0101	a0
승산기 522	W2 = 0011	a1

상기 승산기들은 상기 < 표 19> 에서 보이고 있는 바와 같이 자신에게로 제공되는 입력정보 비트를 월시부호를 구성하는 각각의 심볼들과 승산하여 가산기 540으로 출력한다.

즉, 상기 승산기 520은 a0을 W1 각각의 심볼마다 승산하여 가산기 540으로 출력하고, 상기 승산기 522는 a1을 W2 각각의 심볼마다 승산하여 상기 가산기 540으로 출력한다. 상기 가산기 540은 배타적 가산기를 사용한다.

한편, 승산기 524, 526, 528, 530, 532로는 상기 초기화로 인해 0의 값을 가지는 a2, a3, a4, a5, a6이 각각 인가됨에 따라 상기 월시부호 발생기 500으로부터의 W4, W8, W16과 마스크 발생기 505로부터의 M1, M2에 관계없이 상기 승산기 524, 526, 528, 530, 532의 출력은 상기 가산기 540의 출력에 영향을 미치지 못한다. 즉, 상기 월시부호 생성기 500으로부터 승산기 524, 526, 528로 입력되어지는 W4, W8, W16 심볼열들이 어떤 값이든 상관없이 상기 승산기 524, 526, 528로부터는 0인 심볼열들이 출력된다. 또한, 상기 마스크 생성기 505로부터 승산기 530, 532로 입력되어지는 심볼열들이 어떤 값이든 상관없이 상기 승산기 530, 532로부터는 0인 심볼열들이 출력된다. 따라서, 상기 승산기 524, 526, 528, 530, 532의 출력은 상기 배타적 가산기 540에 입력되어도 아무런 영향을 미치지 않는다. 상기와 같이 a2,a3,a4,a5,a6을 0으로 초기화하는 것은 상기 승산기 524,526, 528, 530, 532의 출력을 차단하는 스위치 동작과 유사하다.

그러면, 상기 배타적 가산기 540은 상기 승산기 520, 522, 524, 526, 528, 530, 532로부터 출력되어진 길이 4인 모든 심볼 열들을 심볼 단위로 배타적 가산하여 반복기 550으로 출력한다.

상기 배타적 가산기 540으로부터 출력되는 길이 4의 부호화 심볼열(Ws)은 하기 < 수학식 13> 로 표현될 수 있다.

수학식 13

$$W_s = (W1 \times 0) + (W2 \times 1)$$

이때, 상기 제어기 510은 상기 반복기 550이 상기 배타적 가산기 540의 출력을 반복할 횟수를 제어하는 제어신호를 출력한다. (24,2) 부호기에서는 상기 배타적 가산기 540으로부터 출력되는 심볼 열이 길이 4을 가짐에 따라 상기 반복기 550의 반복 횟수는 8회가 될 것이다. 따라서, 상기 제어기 510은 상기 반복기 550으로 입력신호 8회 반복(즉, 입력신호를 그대로 출력) 하라는 제어신호를 출력한다. 이에 응답하여 상기 반복기 550은 상기 배타적 가산기 540으로부터 출력되어진 길이 4인 심볼들을 8회 반복한 길이 32의 천공기 560으로 출력한다.

이 시점에서 상기 제어기 510은 천공기 560으로 (24,2) 부호에 대한 천공 위치를 천공하라는 제어신호를 출력한다. 그러면, 상기 천공기 560은 천공위치가 저장되어 있는 메모리 570으로부터 천공위치를 입력받아 상기 반복기 550으로부터 제공되는 길이 32의 부호화 심볼들 중 8개의 해당 부호화 심볼들을 천공한다. 즉, 상기 천공기 560은 상기 반복기 550으로부터 제공되는 길이 32의 부호화 심볼들 중 상기 메모리 570으로부터의 천공위치에 해당하는 8 심볼을 천공한다. 예컨대, 상기 메모리 570으로부터의 천공위치가 0,4,8,12,16,20,24,28인 경우 상기 천공기 560은 상기 길이 32의 부호화 심볼들 중 0,4,8,12,16,20,24,28번째 부호화 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

일곱 번째로, (24,1)부호기로 동작할 때를 살펴보면, 1비트의 데이터를 지시자 a0이 상기 부호기에 입력이 되고, 나머지 데이터를 지시자 a1,a2,a3,a4,a5,a6은 초기화되어 입력된다. 상기 초기화에 의해 상기 a1,a2,a3,a4,a5,a6으로는 0이 입력된다. 한편, 상기 제어기 510으로는 1비트의 데이터를 지시자가 상기 부호기로 입력되어진다는 비트 정보가 입력되어진다. 그러면, 상기 제어기 510은 월시부호 발생기 500으로 길이 2을 가지는 1개의 월시부호를 생성하도록 하는 제어신호를 출력한다. 그러면 상기 월시부호 발생기 500은 길이 2인 월시부호 W1를 출력한다. 상기 월시부호 발생기 500으로부터 발생된 상기 W1는 이에 대응하는 승산기로 제공된다. 예컨대, 상기 W1은 승산기 520으로 제공된다. 상기 월시부호 발생기 500으로부터 발생되는 길이 2의 월시부호는 하기 < 표 20> 과 같다.

[표 20]

월시번호	월시부호
W1	01

한편, 상기 1개의 입력정보 비트들은 상기 승산기들 중 대응하는 각각의 상기 승산기로 제공된다. 상기 1개의 입력정보 비트들과 상기 승산기들의 대응 관계는 하기 < 표 21>에서 보이고 있는 바와 같다.

[표 21]

입력정보 비트	승산기
a0	승산기 520

따라서, 상기 승산기들 각각으로 제공되는 입력정보 비트와 월시부호들은 하기 < 표 22> 과 같이 나타낼 수 있다.

[표 22]

승산기	월시부호	입력정보 비트
승산기 520	W1 = 01	a0

상기 승산기들은 상기 < 표 22> 에서 보이고 있는 바와 같이 자신에게로 제공되는 입력정보 비트를 월시부호를 구성하는 각각의 심볼들과 승산하여 가산기 540으로 출력한다.

즉, 상기 승산기 520은 a0을 W1 각각의 심볼마다 승산하여 가산기 540으로 출력한다. 상기 가산기 540은 배타적 가산기를 사용한다.

한편, 승산기 522, 524, 526, 528, 530, 532로는 상기 초기화로 인해 0의 값을 가지는 a1, a2, a3, a4, a5, a6이 각각 인가됨에 따라 상기 월시부호 발생기 500으로부터의 W2, W4, W8, W16과 마스크 발생기 505로부터의 M1, M2에 관계없이 상기 승산기 522, 524, 526, 528, 530, 532의 출력은 상기 가산기 540의 출력에 영향을 미치지 못한다. 즉, 상기 월시부호 생성기 500으로부터 승산기 522, 524, 526, 528로 입력되어지는 W2, W4, W8, W16 심볼열들이 어떤 값이든 상관없이 상기 승산기 522, 524, 526, 528로부터는 0인 심볼열들이 출력된다. 또한, 상기 마스크 생성기 505로부터 승산기 530, 532로 입력되어지는 심볼열들이 어떤 값이든 상관없이 상기 승산기 530, 532로부터는 0인 심볼열들이 출력된다. 따라서, 상기 승산기 522, 524, 526, 528, 530, 532의 출력은 상기 배타적 가산기 540에 입력되어도 아무런 영향을 미치지 않는다. 상기와 같이 a1, a2, a3, a4, a5, a6을 0으로 초기화하는 것은 상기 승산기 522, 524, 526, 528, 530, 532의 출력을 차단하는 스위치 동작과 유사하다.

그러면, 상기 배타적 가산기 540은 상기 승산기 520, 522, 524, 526, 528, 530, 532로부터 출력되어진 길이 2인 모든 심볼 열들을 심볼 단위로 배타적 가산하여 반복기 550으로 출력한다.

상기 배타적 가산기 540으로부터 출력되는 길이 32의 부호화 심볼열(Ws)은 하기 < 수학식 14> 로 표현될 수 있다.

수학식 14

$$Ws = (W1 \times 0)$$

이때, 상기 제어기 510은 상기 반복기 550이 상기 배타적 가산기 540의 출력을 반복할 횟수를 제어하는 제어신호를 출력한다. (24,1) 부호기에서는 상기 배타적 가산기 540으로부터 출력되는 심볼 열이 길이 2을 가짐에 따라 상기 반복기 550의 반복 횟수는 32회가 될 것이다. 따라서, 상기 제어기 510은 상기 반복기 550으로 입력신호 32회 반복하라는 제어신호를 출력한다. 이에 응답하여 상기 반복기 550은 상기 배타적 가산기 540으로부터 출력되어진 길이 2인 심볼들을 32회 반복한 길이 64의 심볼열을 천공기 560으로 출력한다.

이 시점에서 상기 제어기 510은 천공기 560으로 (24,1) 부호에 대한 천공 위치를 천공하라는 제어신호를 출력한다. 그러면, 상기 천공기 560은 천공위치가 저장되어 있는 메모리 570으로부터 천공위치를 입력받아 상기 반복기 550으로부터 제공되는 길이 64의 부호화 심볼들 중 40개의 해당 부호화 심볼들을 천공한다. 즉, 상기 천공기 560은 상기 반복기 550으로부터 제공되는 길이 64의 부호화 심볼들 중 상기 메모리 570으로부터의 천공위치에 해당하는 40 심볼을 천공

한다. 예컨대, 상기 메모리 570으로부터의 천공위치가 모든 짹수와 1,3,5,7,9,11,13,15인 경우 상기 천공기 560은 상기 길이 64의 부호화 심볼들 중 모든 짹수 번째 부호화 심볼과 1,3,5,7,9,11,13,15번째 부호화 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

제1실시 예(복호기)

도 6은 상기 도 5를 참조하여 살펴본 부호기에 대응하는 복호기의 구조를 도시한다.

이하 본 발명의 실시 예에 따른 복호화 동작을 (24,1) 복호기, (24,2) 복호기, (24,3) 복호기, (24,4) 복호기, (24,5) 복호기, (24,6) 복호기 및 (24,7) 복호기로 구분하여 설명하도록 한다.

첫 번째로, 전술한 (24,7) 부호기에 대응한 복호기의 동작을 살펴보면, 상기 부호기로부터 부호화되어 각각이 + 1 또는 - 1의 값을 가지는 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 수신하고, 상기 수신한 부호화 심볼 열은 0 삽입 기 650으로 입력된다.

한편, 제어기 630은 미리 약속되어졌던 부호 길이의 정보를 입력받아 (24,7) 부호기에 대한 천공위치에 + 1과 - 1의 중간 값인 0을 삽입하라는 제어신호를 출력한다. 상기 0 삽입기 650은 상기 제어기 630으로부터의 제어에 의해 메모리 660으로부터 7비트의 입력 정보비트들에 대응한 천공위치에 대한 정보를 입력받는다. 예컨대, 상기 7비트의 입력 정보비트들에 대응한 천공위치들은 "0,4,8,12,16,20,24,28" 또는 "0,1,2,3,4,5,6,7" 이 된다.

따라서, 상기 0 삽입기 650은 상기 입력받은 부호화 심볼열을 이루는 24개의 부호화 심볼들 중 상기 메모리 660으로부터의 천공위치들에 0을 삽입하여 32개의 심볼들로 이루어진 수신 심볼열을 심볼 누적기 600으로 출력한다. 그러면, 상기 제어기 630은 상기 (24,7) 부호기에 따른 반복 회수만큼 반복되는 심볼을 누적하라는 제어신호를 상기 심볼 누적기 600에 출력한다. 이때, (24,7) 부호기에서 심볼 반복이 없었기 때문에 상기 심볼 누적기 600은 상기 입력받은 32개의 수신 심볼들을 그대로 출력한다. 그러면, 상기 출력되어진 32개의 수신 심볼들은 월시부호와의 상관도 계산기 620으로 입력되고, 이와 동시에 승산기 602, 604, 606으로 각각 입력되어진다. 그러면, 상기 마스크 생성기 610은 길이 32인 마스크 함수 M1을 생성하여 승산기 602로 출력하고, 길이 32인 마스크 함수 M2를 생성하여 승산기 604로 출력한다. 또한, 길이 32인 마스크 함수 M1+ M2를 생성하여 승산기 606으로 출력한다. 이때 상기 마스크 함수는 천공위치에 따라 달라지는데, 각각의 천공위치가 사용되어질 때 부호기에서 사용되어진 마스크 함수들이 사용되어지게 되어진다. 그러면, 상기 승산기 602는 상기 입력된 수신 심볼들과 상기 마스크 함수 M1을 심볼 단위로 승산하여 출력하고, 상기 승산기 604는 상기 입력된 수신 심볼들과 상기 마스크 함수 M2를 심볼 단위로 승산하여 출력한다. 또한, 상기 승산기 602는 상기 입력된 수신 심볼들과 상기 마스크 함수 M1+ M2를 심볼 단위로 승산하여 출력한다. 이 때, 스위치 652는 상기 제어기 630의 제어를 받아 연결이 유지되어 상기 승산기 602에서 출력되어진 심볼열이 월시부호와의 상관도 계산기 622로 입력되도록 한다. 스위치 654는 상기 제어기 630의 제어를 받아 연결이 유지되어 상기 승산기 604에서 출력되어진 심볼열이 월시부호와의 상관도 계산기 624로 입력되도록 한다. 스위치 656은 상기 제어기 630의 제어를 받아 연결이 유지되어 승산기 606에서 출력되어진 심볼열들이 월시부호와의 상관도 계산기 626으로 입력되도록 한다. 그러면, 상기 월시부호와의 상관도 계산기 620은 상기 입력되어진 길이 32인 수신 심볼열과 마스크 함수 M1이 승산되어진 심볼열을 길이 32인 32개의 월시부호들과 상관도를 모두 계산하여, 가장 상관도가 높은 부호에 대한 월시부호 번호, 상관도 값, 그리고, 마스크 번호를 나타내는 0을 상관도 비교기 640으로 출력한다. 상기 월시부호와의 상관도 계산기 622는 상기 입력되어진 길이 32인 수신 심볼열과 마스크 함수 M2가 승산되어진 심볼열을 길이 32인 32개의 월시부호들과 상관도를 모두 계산하여, 가장 상관도가 높은 부호에 대한 월시부호 번호, 상관도 값, 그리고, 마스크 번호를 나타내는 1을 상관도 비교기 640으로 출력한다. 상기 월시부호와의 상관도 계산기 622는 상기 입력되어진 길이 32인 수신 심볼열과 마스크 함수 M1+ M2가 승산되어진 심볼열을 길이 32인 32개의 월시부호들과 상관도를 모두 계산하여, 가장 상관도가 높은 부호에 대한 월시부호 번호, 상관도 값, 그리고, 마스크 번호를 나타내는 2를 상관도 비교기 640으로 출력한다. 상기 월시부호와의 상관도 계산기 622는 상기 입력되어진 길이 32인 수신 심볼열과 마스크 함수 M1+ M2가 승산되어진 심볼열을 길이 32인 32개의 월시부호들과 상관도를 모두 계산하여, 가장 상관도가 높은 부호에 대한 월시부호 번호, 상관도 값, 그리고, 마스크 번호를 나타내는 3을 상관도 비교기 640으로 출력한다. 그러면 상기 상관도 비교기 640은 상기 월시부호와의 상관도 계산기 620, 622, 624, 626으로부터 입력받은 값들 중 상관도 값이 가장 큰 것을 택하고, 그것에 대한 월시번호와 마스크

번호를 연접하여 복호화 비트로 출력한다.

두 번째로, 상기 도 5를 참조하여 살펴본 (24,6) 부호기에 대응한 복호기의 동작을 살펴보면, 상기 부호기로부터 부호화되어 각각이 + 1 또는 - 1의 값을 가지는 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 수신하고, 상기 수신한 부호화 심볼 열은 0 삽입기 650으로 입력된다.

한편, 제어기 630은 미리 약속되어졌던 부호 길이의 정보를 입력받아 (24,6) 부호기에 대한 천공위치에 + 1과 - 1의 종간 값인 0을 삽입하라는 제어신호를 출력한다. 상기 0 삽입기 650은 상기 제어기 630로부터의 제어에 의해 메모리 6 60으로부터 6비트의 입력 정보비트들에 대응한 천공위치에 대한 정보를 입력받는다. 예컨대, 상기 6비트의 입력 정보비트들에 대응한 천공위치들은 " 0,1,2,3,4,5,6,7" 이 된다. 따라서, 상기 0 삽입기 650은 상기 입력받은 부호화 심볼열을 이루는 24개의 부호화 심볼들 중 상기 메모리 660으로부터의 천공위치들에 0을 삽입하여 32개의 심볼들로 이루어진 수신 심볼열을 심볼 누적기 600으로 출력한다. 그러면, 상기 제어기 630은 상기 (24,6) 부호기에 따른 반복 회수만큼 반복되는 심볼을 누적하라는 제어신호를 상기 심볼 누적기 600에 출력한다.

이때, (24,6) 부호기에서 1회 심볼 반복이 있었기 때문에 상기 심볼 누적기 600은 상기 0 삽입기 650으로부터 입력받은 32개의 수신 심볼들을 그대로 출력한다. 그러면, 상기 출력되어진 32개의 수신 심볼들은 월시부호들과의 상관도 계산 620으로 입력되고, 이와 동시에 승산기 602, 604, 606으로 각각 입력되어진다. 그러면, 상기 마스크 생성기 610은 길이 32인 마스크 함수 M1을 생성하여 승산기 602로 출력하고, 길이 32인 마스크 함수 M2를 생성하여 승산기 604로 출력한다. 이때 상기 마스크 함수는 천공위치에 따라 달라지는데, 각각의 천공위치가 사용되어질 때 부호기에서 사용되어진 마스크 함수들이 사용되어지게 되어진다. 그러면, 상기 승산기 602는 상기 입력된 수신 심볼들과 상기 마스크 함수 M1을 심볼 단위로 승산하여 출력하고, 이 때, 스위치 652는 상기 제어기 630의 제어를 받아 연결이 유지되어 상기 승산기 602에서 출력되어진 심볼열이 월시부호와의 상관도 계산기 622로 입력되도록 한다. 스위치 654, 656은 상기 제어기 630의 제어에 의해 연결을 끊기 때문에 상기 승산기 604, 606의 출력은 더 이상 진행되지 않아 상기 승산기 604, 606에서의 동작은 효과 없는 동작을 하게 된다. 그러면, 상기 월시부호와의 상관도 계산기 620은 상기 입력되어진 길이 32인 수신 심볼열을 길이 32인 32개의 월시부호들과 상관도를 모두 계산하여, 가장 상관도가 높은 부호에 대한 월시부호 번호, 상관도 값, 그리고, 마스크 번호를 나타내는 0을 상관도 비교기 640으로 출력한다. 상기 월시부호와의 상관도 계산기 622는 상기 입력되어진 길이 32인 수신 심볼열과 마스크 함수 M1이 승산되어진 심볼열을 길이 32인 32개의 월시부호들과 상관도를 모두 계산하여, 가장 상관도가 높은 부호에 대한 월시부호 번호, 상관도 값, 그리고, 마스크 번호를 나타내는 1을 상관도 비교기 640으로 출력한다. 그러면 상기 상관도 비교기 640은 상기 월시부호와의 상관도 계산기 620으로부터 입력받은 값 중 월시번호와 마스크 번호를 연접하여 복호화 비트로 출력한다.

세 번째로, 상기 도 5를 참조하여 살펴본 (24,5) 부호기에 대응한 복호기의 동작을 살펴보면, 상기 부호기로부터 부호화되어 각각이 + 1 또는 - 1의 값을 가지는 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 수신하고, 상기 수신한 부호화 심볼 열은 0 삽입기 650으로 입력된다.

한편, 제어기 630은 미리 약속되어졌던 부호 길이의 정보를 입력받아 (24,5) 부호기에 대한 천공위치에 + 1과 - 1의 종간 값인 0을 삽입하라는 제어신호를 출력한다. 상기 0 삽입기 650은 상기 제어기 630로부터의 제어에 의해 메모리 6 60으로부터 5비트의 입력 정보비트들에 대응한 천공위치에 대한 정보를 입력받는다. 예컨대, 상기 5비트의 입력 정보비트들에 대응한 천공위치들은 " 0,1,2,3,4,5,6,7" 이 된다. 따라서, 상기 0 삽입기 650은 상기 입력받은 부호화 심볼열을 이루는 24개의 부호화 심볼들 중 상기 메모리 660으로부터의 천공위치들에 0을 삽입하여 32개의 심볼들로 이루어진 수신 심볼열을 심볼 누적기 600으로 출력한다. 그러면, 상기 제어기 630은 상기 (24,5) 부호기에 따른 반복 회수만큼 반복되는 심볼을 누적하라는 제어신호를 상기 심볼 누적기 600에 출력한다.

이때, (24.5) 부호기에서 1회 심볼 반복이 있었기 때문에 상기 심볼 누적기 600은 상기 입력받은 32개의 수신 심볼들을 그대로 출력한다. 그러면, 상기 출력되어진 32개의 수신 심볼들은 월시부호들과의 상관도 계산 620으로 입력되고, 이와 동시에 승산기 602, 604, 606으로 각각 입력되어진다. 이 때, 상기 승산기 602, 604, 606의 출력단의 스위치 6 52, 654, 656은 상기 제어기 630의 제어에 의해 연결을 끊기 때문에 상기 승산기 602, 604, 606의 출력들은 더 이상 진행되지 않아 상기 승산기 602, 604, 606에서의 동작은 효과 없는 동작을 하게 된다. 그러면, 상기 제어기 630은 부호길이와 동일한 월시부호와 상관도를 계산하라는 제어신호(즉, 길이 32인 월시와 상관도 계산하라는 제어신호)를 월시부호와의 상관도 계산기 620에 출력한다. 이 때, 상기 월시부호와의 상관도 계산기 620은 상기 입력되어진 길이 32인 수신 심볼열을 길이 32인 32개의 월시부호들과 상관도를 모두 계산하여, 가장 상관도가 높은 부호에 대한 월시부호 번호, 상관도 값, 그리고, 마스크 번호를 나타내는 값을 상관도 비교기 640으로 출력한다. 그러면 상기 상관도 비교기 640은 상기 월시부호와의 상관도 계산기 620으로부터 입력받은 값 중 월시번호와 마스크 번호를 연접하여 복호화 비트로 출력한다.

네 번째로, 상기 도 5를 참조하여 살펴본 (24.4) 부호기에 대응한 복호기의 동작을 살펴보면, 상기 부호기로부터 부호화되어 각각이 + 1 또는 - 1의 값을 가지는 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 수신하고, 상기 수신한 부호화 심볼 열은 0 삽입기 650으로 입력된다.

한편, 제어기 630은 미리 약속되어졌던 부호 길이의 정보를 입력받아 (24.4) 부호기에 대한 천공위치에 + 1과 - 1의 종간 값인 0을 삽입하라는 제어신호를 출력한다. 상기 0 삽입기 650은 상기 제어기 630로부터의 제어에 의해 메모리 6 60으로부터 4비트의 입력 정보비트들에 대응한 천공위치에 대한 정보를 입력받는다. 예컨대, 상기 4비트의 입력 정보비트들에 대응한 천공위치들은 "0, 1, 2, 3, 4, 5, 16" 이 될 수 있다. 따라서, 상기 0 삽입기 650은 상기 입력받은 부호화 심볼열을 이루는 24개의 부호화 심볼들 중 상기 메모리 660으로부터의 천공위치들에 0을 삽입하여 32개의 심볼들로 이루어진 수신 심볼열을 심볼 누적기 600으로 출력한다. 그러면, 상기 제어기 630은 상기 (24.4) 부호기에 따른 반복 회수만큼 반복되는 심볼을 누적하라는 제어신호를 상기 심볼 누적기 600에 출력한다.

이때, (24.4) 부호기에서 2회의 심볼 반복이 있었기 때문에 상기 심볼 누적기 660은 상기 입력받은 32개의 수신 심볼들 중 반복 위치에 있는 2심볼씩을 더해서 길이 16인 심볼열들을 출력한다. 그러면, 상기 출력되어진 16개의 수신 심볼들은 월시부호들과의 상관도 계산 620으로 입력되고, 이와 동시에 승산기 602, 604, 606으로 각각 입력되어진다. 이 때, 상기 승산기 602, 604, 606의 출력단의 스위치 6 52, 654, 656은 상기 제어기 630의 제어에 의해 연결을 끊기 때문에 상기 승산기 602, 604, 606의 출력들은 더 이상 진행되지 않아 상기 승산기 602, 604, 606에서의 동작은 효과 없는 동작을 하게 된다. 그러면, 상기 제어기 630은 부호길이와 동일한 월시부호와 상관도를 계산하라는 제어신호(즉, 길이 16인 월시와 상관도 계산하라는 제어신호)를 월시부호와의 상관도 계산기 620에 출력한다. 이 때, 상기 월시부호와의 상관도 계산기 620은 상기 입력되어진 길이 16인 수신 심볼열을 길이 16인 16개의 월시부호들과 상관도를 모두 계산하여, 가장 상관도가 높은 부호에 대한 월시부호 번호, 상관도 값, 그리고, 마스크 번호를 나타내는 값을 상관도 비교기 640으로 출력한다. 그러면 상기 상관도 비교기 640은 상기 월시부호와의 상관도 계산기 620으로부터 입력받은 값 중 월시번호와 마스크 번호를 연접하여 복호화 비트로 출력한다.

다섯 번째로, 상기 도 5를 참조하여 살펴본 (24.3) 부호기에 대응한 복호기의 동작을 살펴보면, 상기 부호기로부터 부호화되어 각각이 + 1 또는 - 1의 값을 가지는 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 수신하고, 상기 수신한 부호화 심볼 열은 0 삽입기 650으로 입력된다.

한편, 제어기 630은 미리 약속되어졌던 부호 길이의 정보를 입력받아 (24.3) 부호기에 대한 천공위치에 + 1과 - 1의 종간 값인 0을 삽입하라는 제어신호를 출력한다. 상기 0 삽입기 650은 상기 제어기 630로부터의 제어에 의해 메모리 6 60으로부터 3비트의 입력 정보비트들에 대응한 천공위치에 대한 정보를 입력받는다. 예컨대, 상기 3비트의 입력 정보

비트들에 대응한 천공위치들은 " 0,3,5,6,7,8,16,24" 이 될 수 있다. 따라서, 상기 0 삽입기 650은 상기 입력받은 부호화 심볼열을 이루는 24개의 부호화 심볼들 중 상기 메모리 660으로부터의 천공위치들에 0을 삽입하여 32개의 심볼들로 이루어진 수신 심볼열을 심볼 누적기 600으로 출력한다. 그러면, 상기 제어기 630은 상기 (24,3) 부호기에 따른 반복 회수만큼 반복되는 심볼을 누적하라는 제어신호를 상기 심볼 누적기 600에 출력한다.

이때, (24,3) 부호기에서 4회의 심볼 반복이 있었기 때문에 상기 심볼 누적기 660은 상기 입력받은 32개의 수신 심볼들 중 반복 위치에 있는 4심볼씩을 더해서 길이 8인 심볼열들을 출력한다. 그러면, 상기 출력되어진 8개의 수신 심볼들은 월시부호들과의 상관도 계산 620으로 입력되고, 이와 동시에 승산기 602, 604, 606으로 각각 입력되어진다. 이 때, 상기 승산기 602, 604, 606의 출력단의 스위치 652, 654, 656은 상기 제어기 630의 제어에 의해 연결을 끊기 때문에 상기 승산기 602, 604, 606의 출력들은 더 이상 진행되지 않아 상기 승산기 602, 604, 606에서의 동작은 효과 없는 동작을 하게 된다. 그러면, 상기 제어기 630은 부호길이와 동일한 월시부호와 상관도를 계산하라는 제어신호(즉, 길이 8인 월시와 상관도 계산하라는 제어신호)를 월시부호와의 상관도 계산기 620에 출력한다. 이 때, 상기 월시부호와의 상관도 계산기 620은 상기 입력되어진 길이 8인 수신 심볼열을 길이 8인 8개의 월시부호들과 상관도를 모두 계산하여, 가장 상관도가 높은 부호에 대한 월시부호 번호, 상관도 값, 그리고, 마스크 번호를 나타내는 값을 상관도 비교기 640으로 출력한다. 그러면 상기 상관도 비교기 640은 상기 월시부호와의 상관도 계산기 620으로부터 입력받은 값 중 월시 번호와 마스크 번호를 연접하여 복호화 비트로 출력한다.

여섯 번째로, 상기 도 5를 참조하여 살펴본 (24,2) 부호기에 대응한 복호기의 동작을 살펴보면, 상기 부호기로부터 부호화되어 각각이 + 1 또는 - 1의 값을 가지는 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 수신하고, 상기 수신한 부호화 심볼 열은 0 삽입기 650으로 입력된다.

한편, 제어기 630은 미리 약속되어졌던 부호 길이의 정보를 입력받아 (24,2) 부호기에 대한 천공위치에 + 1과 - 1의 중간 값인 0을 삽입하라는 제어신호를 출력한다. 상기 0 삽입기 650은 상기 제어기 630로부터의 제어에 의해 메모리 660으로부터 2비트의 입력 정보비트들에 대응한 천공위치에 대한 정보를 입력받는다. 예컨대, 상기 2비트의 입력 정보비트들에 대응한 천공위치들은 " 0,4,8,12,16,20,24,28" 이 될 수 있다. 따라서, 상기 0 삽입기 650은 상기 입력받은 부호화 심볼열을 이루는 24개의 부호화 심볼들 중 상기 메모리 660으로부터의 천공위치들에 0을 삽입하여 32개의 심볼들로 이루어진 수신 심볼열을 심볼 누적기 600으로 출력한다. 그러면, 상기 제어기 630은 상기 (24,2) 부호기에 따른 반복 회수만큼 반복되는 심볼을 누적하라는 제어신호를 상기 심볼 누적기 600에 출력한다.

이때, (24,2) 부호기에서 8회의 심볼 반복이 있었기 때문에 상기 심볼 누적기 660은 상기 입력받은 32개의 수신 심볼들 중 반복 위치에 있는 8심볼씩을 더해서 길이 4인 심볼열들을 출력한다. 그러면, 상기 출력되어진 4개의 수신 심볼들은 월시부호들과의 상관도 계산 620으로 입력되고, 이와 동시에 승산기 602, 604, 606으로 각각 입력되어진다. 이 때, 상기 승산기 602, 604, 606의 출력단의 스위치 652, 654, 656은 상기 제어기 630의 제어에 의해 연결을 끊기 때문에 상기 승산기 602, 604, 606의 출력들은 더 이상 진행되지 않아 상기 승산기 602, 604, 606에서의 동작은 효과 없는 동작을 하게 된다. 그러면, 상기 제어기 630은 부호길이와 동일한 월시부호와 상관도를 계산하라는 제어신호(즉, 길이 4인 월시와 상관도 계산하라는 제어신호)를 월시부호와의 상관도 계산기 620에 출력한다. 이 때, 상기 월시부호와의 상관도 계산기 620은 상기 입력되어진 길이 4인 수신 심볼열을 길이 4인 4개의 월시부호들과 상관도를 모두 계산하여, 가장 상관도가 높은 부호에 대한 월시부호 번호, 상관도 값, 그리고, 마스크 번호를 나타내는 값을 상관도 비교기 640으로 출력한다. 그러면 상기 상관도 비교기 640은 상기 월시부호와의 상관도 계산기 620으로부터 입력받은 값 중 월시 번호와 마스크 번호를 연접하여 복호화 비트로 출력한다.

일곱 번째로, 상기 도 5를 참조하여 살펴본 (24,1) 부호기에 대응한 복호기의 동작을 살펴보면, 상기 부호기로부터 부호화되어 각각이 + 1 또는 - 1의 값을 가지는 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 수신하고, 상기 수신한 부호화 심볼 열은 0 삽입기 650으로 입력된다.

한편, 제어기 630은 미리 약속되어졌던 부호 길이의 정보를 입력받아 (24,1)부호기에 대한 천공위치에 +1과 -1의 중간 값인 0을 삽입하라는 제어신호를 출력한다. 상기 0 삽입기 650은 상기 제어기 630로부터의 제어에 의해 메모리 6 60으로부터 1비트의 입력 정보비트들에 대응한 천공위치에 대한 정보를 입력받는다. 예컨대, 상기 1비트의 입력 정보비트들에 대응한 천공위치들은 "모든 짹수와 1,3,5,7,9,11,13,15" 가 될 수 있다. 따라서, 상기 0 삽입기 650은 상기 입력받은 부호화 심볼열을 이루는 24개의 부호화 심볼들 중 상기 메모리 660으로부터의 천공위치들에 0을 삽입하여 64개의 심볼들로 이루어진 수신 심볼열을 심볼 누적기 600으로 출력한다. 그러면, 상기 제어기 630은 상기 (24,1)부호기에 따른 반복 회수만큼 반복되는 심볼을 누적하라는 제어신호를 상기 심볼 누적기 600에 출력한다.

이때, (24,1)부호기에서 32회의 심볼 반복이 있었기 때문에 상기 심볼 누적기 660은 상기 입력받은 64개의 수신 심볼들 중 반복 위치에 있는 32심볼씩을 더해서 길이 2인 심볼열들을 출력한다. 그러면, 상기 출력되어진 2개의 수신 심볼들은 월시부호들과의 상관도 계산 620으로 입력되고, 이와 동시에 승산기 602, 604, 606으로 각각 입력되어진다. 이 때, 상기 승산기 602, 604, 606의 출력단의 스위치 652, 654, 656은 상기 제어기 630의 제어에 의해 연결을 끊기 때문에 상기 승산기 602, 604, 606의 출력들은 더 이상 진행되지 않아 상기 승산기 602, 604, 606에서의 동작은 효과 없는 동작을 하게 된다. 그러면, 상기 제어기 630은 부호길이와 동일한 월시부호와의 상관도를 계산하라는 제어신호(즉, 길이 2인 월시와 상관도 계산하라는 제어신호)를 월시부호와의 상관도 계산기 620에 출력한다. 이 때, 상기 월시부호와의 상관도 계산기 620은 상기 입력되어진 길이 2인 수신 심볼열을 길이 2인 2개의 월시부호들과 상관도를 모두 계산하여, 가장 상관도가 높은 부호에 대한 월시부호 번호, 상관도 값, 그리고, 마스크 번호를 나타내는 값을 상관도 비교기 640으로 출력한다. 그러면 상기 상관도 비교기 640은 상기 월시부호와의 상관도 계산기 620으로부터 입력받은 값 중 월시번호와 마스크 번호를 연결하여 복호화 비트로 출력한다.

제 2 실시 예(부호기)

도 9는 서로 다른 길이인 (24,1)부호기, (24,2)부호기, (24,3)부호기, (24,4)부호기, (24,5)부호기, (24,6)부호기 및 (24,7)부호기를 모두 수행할 수 있는 부호기의 또 다른 구조를 나타낸다. 즉, 상기 도 9에서는 1~7비트의 입력 정보비트들을 입력하고, 상기 1~7비트의 입력 정보비트들 각각을 길이 32의 서로 다른 월시부호들 또는 마스크들로 부호화하여 24개의 부호화 심볼들로 이루어진 부호화 심볼열로 출력하는 이동통신시스템에서의 부호기의 구성을 보이고 있다. 이때, 상기 제1실시 예와 달리 부호기에 있어서, 심볼 반복기가 없이 구성되어질 수 있는 구조를 나타낸다.

상기 도 9를 참조하여 본 발명의 다른 실시 예에 따른 부호기의 구성을 살펴보면, 제어기 1110은 상기 입력 정보비트들로 입력된 비트 수를 검사하여 상기 부호화를 위한 제어를 수행한다. 1~7비트의 입력 정보비트가 입력되었을 경우 천공기 1160으로 상기 입력 정보비트들에 대응한 8개의 천공 위치들을 천공하도록 제어한다.

상기 월시부호 발생기 1100은 길이 32의 월시부호들을 출력한다. 예컨대, 입력 정보비트들이 입력되면 상기 월시부호 발생기 1100은 길이 32의 서로 다른 5개의 월시부호들을 출력한다.

상기 마스크 발생기 1105는 길이 32의 마스크들을 출력한다. 예컨대, 입력 정보비트들이 입력되면 상기 마스크 발생기 1105는 길이 32의 서로 다른 2개의 마스크들을 출력한다.

승산기들 1120 내지 1132는 상기 1~7비트의 입력 정보비트들과 상기 월시부호 발생기 1100과 상기 마스크 발생기 1105로부터의 월시부호들 및 마스크들을 일대일로 승산하여 32개의 부호화 심볼들로 이루어진 부호화 심볼열들을 출력한다. 배타적 가산기 1140은 상기 승산기들 1120 내지 1132로부터의 부호화 심볼열들을 배타적 가산하여 32개의 부호화 심볼들로 이루어지는 하나의 부호화 심볼열을 출력한다. 배타적 가산기 1140에서 출력된 길이 32의 부호화 심볼열은 천공기 1160에 입력된다.

메모리 1170은 상기 1비트의 입력 정보비트들에 대응한 8개의 천공 위치들, 2비트의 입력 정보비트들에 대응한 8개의 천공 위치들, 3비트의 입력 정보비트들에 대응한 8개의 천공 위치들, 3비트의 입력 정보비트들에 대응한 8개의 천공 위치들, 4비트의 입력 정보비트들에 대응한 8개의 천공 위치들, 5비트의 입력 정보비트들에 대응한 8개의 천공 위치들, 6비트의 입력 정보비트들에 대응한 8개의 천공 위치들 및 상기 7비트의 입력 정보비트들에 대응한 8개의 천공 위치들을 저장한다. 상기 1비트의 입력 정보비트들에 대응하여 상기 메모리 1170에 저장된 8개의 천공 위치들은 1,3,5,7,9, 11,13,15번째 심볼들이며, 2비트의 입력 정보비트들에 대응하여 상기 메모리 1170에 저장된 8개의 천공 위치들은 0, 4,8,12,16,20,24,28번째 심볼들이며, 3비트의 입력 정보비트들에 대응하여 상기 메모리 1170에 저장된 8개의 천공 위치들은 0,3,5,6,7,8,16,24번째 심볼들이며, 4비트의 입력 정보비트들에 대응하여 상기 메모리 1170에 저장된 8개의 천공 위치들은 0,1,2,3,4,5,6,16번째 심볼들이며, 5비트의 입력 정보비트들에 대응하여 상기 메모리 1170에 저장된 8개의 천공 위치들은 0,1,2,3,4,5,6,7번째 심볼들이며, 6비트의 입력 정보비트들에 대응하여 상기 메모리 1170에 저장된 8개의 천공 위치들은 0,1,2,3,4,5,6,7번째 심볼들이며, 7비트의 입력 정보비트들에 대응하여 상기 메모리 1170에 저장된 8개의 천공 위치들은 0,1,2,3,4,5,6,7번째 심볼들이다.

상기한 입력 정보비트들에 대응하는 천공 위치들을 표로서 보이면 하기 < 표 23> 와 같다.

[표 23]

입력 정보비트 수	천공 위치
1	1,3,5,7,9,11,13,15번째 심볼
2	0,4,8,12,16,20,24,28번째 심볼
3	0,3,5,6,7,8,16,24번째 심볼
4	0,1,2,3,4,5,6,16번째 심볼
5	0,1,2,3,4,5,6,7번째 심볼
6	0,1,2,3,4,5,6,7번째 심볼
7	0,1,2,3,4,5,6,7번째 심볼

상기 천공기 1160은 상기 배타적 가산기 1140으로부터의 상기 32개의 부호화 심볼들로 이루어진 부호화 심볼열을 입력하고, 상기 제어기 1110으로부터의 제어에 의해 상기 32개의 부호화 심볼들 중 상기 메모리 1170으로부터 출력한 8개의 해당 천공 위치들의 부호화 심볼들을 천공하여 상기 24개의 부호화 심볼들로 이루어진 상기 부호화 심볼열을 출력한다. 즉, 상기 천공기 1160은 1비트의 입력 정보비트들이 입력되면 상기 제어기 1110의 제어에 의해 상기 배타적 가산기 1140으로부터 출력되는 32개의 부호화 심볼들 중 1,3,5,7,9,11,13,15번째 심볼들을 천공하여 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 출력한다. 또한, 상기 천공기 1160은 2비트의 입력 정보비트들이 입력되면 상기 제어기 1110의 제어에 의해 상기 배타적 가산기 1140으로부터 출력되는 32개의 부호화 심볼들 중 0,4,8,12,16,20,24,28번째 심볼들을 천공하여 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 출력한다. 즉, 상기 천공기 1160은 3비트의 입력 정보비트들이 입력되면 상기 제어기 1110의 제어에 의해 상기 배타적 가산기 1140으로부터 출력되는 32개의 부호화 심볼들 중 0,3,5,6,7,8,16,24번째 심볼들을 천공하여 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 출력한다. 즉, 상기 천공기 1160은 4비트의 입력 정보비트들이 입력되면 상기 제어기 1110의 제어에 의해 상기 배타적 가산기 1140으로부터 출력되는 32개의 부호화 심볼들 중 0,1,2,3,4,5,6,16번째 심볼들을 천공하여 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 출력한다. 즉, 상기 천공기 1160은 5비트의 입력 정보비트들이 입력되면 상기 제어기 1110의 제어에 의해 상기 배타적 가산기 1140으로부터 출력되는 32개의 부호화 심볼들 중 0,1,2,3,4,5,6,7번째 심볼들을 천공하여 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 출력한다. 즉, 상기 천공기 1160은 6비트의 입력 정보비트들이 입력되면 상기 제어기 1110의 제어에 의해 상기 배타적 가산기 1140으로부터 출력되는 32개의 부호화 심볼들 중 0,1,2,3,4,5,6,7번째 심볼들을 천공하여 24개의 부호화 심볼들로 이루어진 부호화 심볼열을 출력한다. 또한, 상기 천공기 1160은 7비트의 입력 정보비트들이 입력되면 상기 제어기 1110의 제어에 의해 상기 배타적 가산기 1140으로부터 출력되는 32개의 부호화 심볼들 중 0,1,2,3,4,5,6,7번째 심볼들을 천공하여 24개의 부호화 심볼들로 이

루어진 부호화 심볼열을 출력한다.

먼저 상술한 도 9를 참조하여 상기 부호기를 설명하기 위해서, (24,1) 부호기로 동작할 때, (24,2) 부호기로 동작할 때, (24,3) 부호기로 동작할 때, (24,4) 부호기로 동작할 때, (24,5) 부호기로 동작할 때, (24,6) 부호기로 동작할 때 및 (24,7) 부호기로 동작할 때를 나누어 설명하기로 한다.

첫 번째로, (24,7) 부호기로의 동작을 살펴보면, 7비트의 데이터를 지시자 a0,a1,a2,a3,a4,a5,a6이 상기 부호기에 입력이 되면, 상기 7비트의 데이터를 지시자가 상기 부호기로 입력되어진다는 비트 정보가 제어기 1110으로 입력되어진다. 그러면, 상기 제어기 1110은 월시부호 발생기 1100으로 길이 32를 가지는 월시부호를 생성하도록 하는 제어신호를 출력한다. 그러면 상기 월시부호 발생기 1100은 길이 32인 월시부호 W1, W2, W4, W8, W16을 출력한다. 상기 월시부호 발생기 1100으로부터 발생된 상기 W1, W2, W4, W8, W16은 각각에 대응하는 승산기로 제공된다. 예컨대, 상기 W1은 승산기 1120으로 제공되며, 상기 W2는 승산기 1122로 제공된다. 상기 W4는 승산기 1124로 제공되며, 상기 W8은 승산기 1126으로 제공된다. 마지막으로 상기 W16은 승산기 1128로 제공된다. 상기 월시부호 발생기 1100으로부터 발생되는 길이 32의 월시부호는 하기 < 표 24> 과 같다.

[표 24]

월시번호	월시부호
W1	0101 0101 0101 0101 0101 0101 0101 0101
W2	0011 0011 0011 0011 0011 0011 0011 0011
W4	0000 1111 0000 1111 0000 1111 0000 1111
W8	0000 0000 1111 1111 0000 0000 1111 1111
W16	0000 0000 0000 0000 1111 1111 1111 1111

이와 동시에, 마스크 발생기 1105는 마스크 $M1 = 0000\ 0000\ 0000\ 0000\ 1111\ 1111\ 1111\ 1111$ 을 발생시켜 승산기 1130으로 출력하며, 마스크 $M2 = 0000\ 0000\ 1110\ 1000\ 1101\ 1000\ 1100\ 0000$ 을 발생시켜 승산기 1132로 출력한다.

한편, 상기 7개의 입력정보 비트들은 상기 승산기들 중 대응하는 각각의 상기 승산기로 제공된다. 상기 7개의 입력정보 비트들과 상기 승산기들의 대응 관계는 하기 < 표 25>에서 보이고 있는 바와 같다.

[표 25]

입력정보 비트	승산기
a0	승산기 1120
a1	승산기 1122
a2	승산기 1124
a3	승산기 1126
a4	승산기 1128
a5	승산기 1130
a6	승산기 1132

따라서, 상기 승산기들 각각으로 제공되는 입력정보 비트와 월시부호 및 마스크는 하기 < 표 26>과 같이 나타낼 수 있다.

[표 26]

승산기	월시부호/마스크	입력정보 비트
승산기 1120	W1 = 0101 0101 0101 0101 0101 0101 0101 0101	a0
승산기 1122	W2 = 0011 0011 0011 0011 0011 0011 0011 0011	a1
승산기 1124	W4 = 0000 1111 0000 1111 0000 1111 0000 1111	a2
승산기 1126	W8 = 0000 0000 1111 1111 0000 0000 1111 1111	a3
승산기 1128	W16 = 0000 0000 0000 0000 1111 1111 1111 1111	a4
승산기 1130	M1 = 0000 0000 0000 0000 1111 1111 1111 1111	a5
승산기 1132	M2 = 0000 0000 1110 1000 1101 1000 1100 0000	a6

상기 승산기들은 상기 < 표 26> 에서 보이고 있는 바와 같이 자신에게로 제공되는 입력정보 비트를 월시부호/마스크를 구성하는 각각의 심볼들과 승산하여 가산기 1140으로 출력한다.

즉, 상기 승산기 1120은 a0을 W1 각각의 심볼마다 승산하여 가산기 1140으로 출력하고, 상기 승산기 1122는 a1을 W2 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력하며, 상기 승산기 1124는 a2를 W4 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력한다. 한편, 상기 승산기 1126은 a3을 W8 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력하고, 상기 승산기 1128은 a4를 W16 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력한다. 또한, 상기 승산기 1130은 a5를 M1 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력하고, 상기 승산기 1132는 a6를 M2 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력한다. 상기 가산기 1140은 배타적 가산기를 사용한다.

그러면, 상기 배타적 가산기 1140은 상기 승산기 1120, 1122, 1124, 1126, 1128, 1130, 1132로부터 출력되어진 길이 32인 모든 심볼 열들을 심볼 단위로 배타적 가산하여 반복기 1150으로 출력한다.

상기 배타적 가산기 1140으로부터 출력되는 길이 32의 부호화 심볼열(Ws)은 하기 < 수학식 15> 로 표현될 수 있다.

수학식 15

$$Ws = (W1 \times 0) + (W2 \times 1) + (W4 \times 2) + (W8 \times 3) + (W16 \times 4) + (M1 \times 5) + (M2 \times 6)$$

이때, 상기 제어기 1110은 천공기 1160으로 (24,7) 부호에 대한 천공 위치를 천공하라는 제어신호를 출력한다. 그러면, 상기 천공기 1160은 천공위치가 저장되어 있는 메모리 1170으로부터 천공위치를 입력받아 상기 배타적 가산기 1140으로부터 제공되는 길이 32의 부호화 심볼들 중 8개의 해당 부호화 심볼들을 천공한다. 즉, 상기 천공기 1160은 상기 배타적 가산기 1140으로부터 제공되는 길이 32의 부호화 심볼들 중 상기 메모리 1170으로부터의 천공위치에 해당하는 8 심볼을 천공한다. 예컨대, 상기 메모리 1170으로부터의 천공위치가 0,1,2,3,4,5,6,7인 경우 상기 천공기 1160은 상기 길이 32의 부호화 심볼들 중 0,1,2,3,4,5,6,7번째 부호화 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

두 번째로, (24,6) 부호기로의 동작을 살펴보면, 6비트의 데이터를 지시자 a0,a1,a2,a3,a4,a5이 상기 부호기에 입력이 되면, 상기 6비트의 데이터를 지시자가 상기 부호기로 입력되어진다는 비트 정보가 제어기 1110으로 입력되어진다. 그러면, 상기 제어기 1110은 월시부호 발생기 1100으로 길이 32를 가지는 월시부호를 생성하도록 하는 제어신호를 출력한다. 그러면 상기 월시부호 발생기 1100은 길이 32인 월시부호 W1, W2, W4, W8, W16을 출력한다. 상기 월시부호 발생기 1100으로부터 발생된 상기 W1, W2, W4, W8, W16은 각각에 대응하는 승산기로 제공된다. 예컨대, 상기 W1은 승산기 1120으로 제공되며, 상기 W2는 승산기 1122로 제공된다. 상기 W4는 승산기 1124로 제공되며, 상기 W8은 승산기 1126으로 제공된다. 마지막으로 상기 W16은 승산기 1128로 제공된다. 상기 월시부호 발생기 1100으로부터 발생되는 길이 32의 월시부호는 하기 < 표 27> 과 같다.

[표 27]

월시번호	월시부호
W1	0101 0101 0101 0101 0101 0101 0101 0101
W2	0011 0011 0011 0011 0011 0011 0011 0011
W4	0000 1111 0000 1111 0000 1111 0000 1111
W8	0000 0000 1111 1111 0000 0000 1111 1111
W16	0000 0000 0000 0000 1111 1111 1111 1111

이와 동시에, 마스크 발생기 1105는 마스크 $M1 = 0111 0111 0010 0100 0110 0000 0000 0000$ 을 발생시켜 승산기 1130으로 출력한다.

한편, 상기 6개의 입력정보 비트들은 상기 승산기들 중 대응하는 각각의 상기 승산기로 제공된다. 상기 6개의 입력정보 비트들과 상기 승산기들의 대응 관계는 하기 < 표 28> 에서 보이고 있는 바와 같다.

[표 28]

입력정보 비트	승산기
a0	승산기 1120
a1	승산기 1122
a2	승산기 1124
a3	승산기 1126
a4	승산기 1128
a5	승산기 1130

따라서, 상기 승산기들 각각으로 제공되는 입력정보 비트와 월시부호 및 마스크는 하기 < 표 29> 과 같이 나타낼 수 있다.

[표 29]

승산기	월시부호/마스크	입력정보 비트
승산기 1120	$W1 = 0101 0101 0101 0101 0101 0101 0101 0101$	a0
승산기 1122	$W2 = 0011 0011 0011 0011 0011 0011 0011 0011$	a1
승산기 1124	$W4 = 0000 1111 0000 1111 0000 1111 0000 1111$	a2
승산기 1126	$W8 = 0000 0000 1111 1111 0000 0000 1111 1111$	a3
승산기 1128	$W16 = 0000 0000 0000 0000 1111 1111 1111 1111$	a4
승산기 1130	$M1 = 0111 0111 0010 0100 0110 0000 0000 0000$	a5

상기 승산기들은 상기 < 표 29> 에서 보이고 있는 바와 같이 자신에게로 제공되는 입력정보 비트를 월시부호/마스크를 구성하는 각각의 심볼들과 승산하여 가산기 1140으로 출력한다.

즉, 상기 승산기 1120은 a0를 W1 각각의 심볼마다 승산하여 가산기 1140으로 출력하고, 상기 승산기 1122는 a1을 W2 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력하며, 상기 승산기 1124는 a2를 W4 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력한다. 한편, 상기 승산기 1126은 a3을 W8 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력하고, 상기 승산기 1128은 a4를 W16 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력한다. 또한, 상기 승산기 1130은 a5를 M1 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력한다. 상기 가산기 1140은 배타적 가산기를 사용한다.

한편, 승산기 1132로는 상기 초기화로 인해 0의 값을 가지는 a6이 각각 인가됨에 따라 상기마스크 발생기 1105로부터의 M2에 관계없이 상기 승산기 1132의 출력은 상기 가산기 1140의 출력에 영향을 미치지 못한다.

즉, 상기 마스크 생성기 1105로부터 승산기 1132로 입력되어지는 심볼열들이 어떤 값이든 상관없이 상기 승산기 1132로부터는 0인 심볼열들이 출력된다. 따라서, 상기 승산기 1132의 출력은 상기 배타적 가산기 1140에 입력되어도 아무런 영향을 미치지 않는다. 상기와 같이 a6을 0으로 초기화하는 것은 상기 승산기 1132의 출력을 차단하는 스위치 동작과 유사하다.

그러면, 상기 배타적 가산기 1140은 상기 승산기 1120, 1122, 1124, 1126, 1128, 1130, 1132로부터 출력되어진 길이 32인 모든 심볼 열들을 심볼 단위로 배타적 가산하여 천공기 1160으로 출력한다.

상기 배타적 가산기 1140으로부터 출력되는 길이 32의 부호화 심볼열(Ws)은 하기 < 수학식 16>로 표현될 수 있다.

수학식 16

$$Ws = (W1 \times 0) + (W2 \times 1) + (W4 \times 2) + (W8 \times 3) + (W16 \times 4) + (M1 \times 5)$$

이때, 상기 제어기 1110은 천공기 1160으로 (24,6) 부호에 대한 천공 위치를 천공하라는 제어신호를 출력한다. 그러면, 상기 천공기 1160은 천공위치가 저장되어 있는 메모리 1170으로부터 천공위치를 입력받아 상기 배타적 가산기 1140으로부터 제공되는 길이 32의 부호화 심볼들 중 8개의 해당 부호화 심볼들을 천공한다. 즉, 상기 천공기 1160은 상기 배타적 가산기 1140으로부터 제공되는 길이 32의 부호화 심볼들 중 상기 메모리 1170으로부터의 천공위치에 해당하는 8 심볼을 천공한다. 예컨대, 상기 메모리 1170으로부터의 천공위치가 0,4,8,12,16,20,24,28인 경우 상기 천공기 1160은 상기 길이 32의 부호화 심볼들 중 0,4,12,16,20,24,28번째 부호화 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

상기 (24,6) 부호기의 동작은 천공위치로 {0,4,8,12,16,20,24,28}을 가질 때에 대한 구성을 나타냈다. 하지만, 상기 (24,6) 부호기는 하드웨어의 복잡도를 줄이기 위해 {0,1,2,3,4,5,6,7}을 천공위치로서 가질 수 있다. 이와 같이 {0,1,2,3,4,5,6,7}을 천공위치로 하는 (24,6) 부호기의 경우 마스크 발생기 1105는 마스크 M1로 "0000 0000 1110 1000 1101 1000 1100 0000"을 발생한다. 예컨대, 상기 메모리 1170으로부터의 천공위치가 0,1,2,3,4,5,6,7인 경우 상기 천공기 1160은 상기 길이 32의 부호화 심볼들 중 0,1,2,3,4,5,6,7번째 부호화 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

세 번째로, (24,5) 부호기로의 동작을 살펴보면, 5비트의 데이터를 지시자 a0,a1,a2,a3,a4이 상기 부호기에 입력이 되면, 상기 5비트의 데이터를 지시자가 상기 부호기로 입력되어진다는 비트 정보가 제어기 1110으로 입력되어진다. 그러면, 상기 제어기 1110은 월시부호 발생기 1100으로 길이 32를 가지는 월시부호를 생성하도록 하는 제어신호를 출력한다. 그러면 상기 월시부호 발생기 1100은 길이 32인 월시부호 W1, W2, W4, W8, W16을 출력한다. 상기 월시부호 발생기 1100으로부터 발생된 상기 W1, W2, W4, W8, W16은 각각에 대응하는 승산기로 제공된다. 예컨대, 상기 W1은 승산기 1120으로 제공되며, 상기 W2는 승산기 1122로 제공된다. 상기 W4는 승산기 1124로 제공되며, 상기 W8은 승산기 1126으로 제공된다. 마지막으로 상기 W16은 승산기 1128로 제공된다. 상기 월시부호 발생기 1100으로부터 발생되는 길이 32의 월시부호는 하기 < 표 30>과 같다.

[표 30]

월시번호	월시부호
W1	0101 0101 0101 0101 0101 0101 0101 0101
W2	0011 0011 0011 0011 0011 0011 0011 0011
W4	0000 1111 0000 1111 0000 1111 0000 1111
W8	0000 0000 1111 1111 0000 0000 1111 1111
W16	0000 0000 0000 0000 1111 1111 1111 1111

한편, 상기 5개의 입력정보 비트들은 상기 승산기들 중 대응하는 각각의 상기 승산기로 제공된다. 상기 5개의 입력정보 비트들과 상기 승산기들의 대응 관계는 하기 < 표 31> 에서 보이고 있는 바와 같다.

[표 31]

입력정보 비트	승산기
a0	승산기 1120
a1	승산기 1122
a2	승산기 1124
a3	승산기 1126
a4	승산기 1128

따라서, 상기 승산기들 각각으로 제공되는 입력정보 비트와 월시부호는 하기 < 표 32> 과 같이 나타낼 수 있다.

[표 32]

승산기	월시부호	입력정보 비트
승산기 1120	W1 = 0101 0101 0101 0101 0101 0101 0101 0101	a0
승산기 1122	W2 = 0011 0011 0011 0011 0011 0011 0011 0011	a1
승산기 1124	W4 = 0000 1111 0000 1111 0000 1111 0000 1111	a2
승산기 1126	W8 = 0000 0000 1111 1111 0000 0000 1111 1111	a3
승산기 1128	W16 = 0000 0000 0000 0000 1111 1111 1111 1111	a4

상기 승산기들은 상기 < 표 32> 에서 보이고 있는 바와 같이 자신에게로 제공되는 입력정보 비트를 월시부호를 구성하는 각각의 심볼들과 승산하여 가산기 1140으로 출력한다.

즉, 상기 승산기 1120은 a0을 W1 각각의 심볼마다 승산하여 가산기 1140으로 출력하고, 상기 승산기 1122는 a1을 W2 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력하며, 상기 승산기 1124는 a2를 W4 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력한다. 한편, 상기 승산기 1126은 a3을 W8 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력하고, 상기 승산기 1128은 a4를 W16 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력한다. 상기 가산기 1140은 배타적 가산기를 사용한다.

한편, 승산기 1130, 1132로는 상기 초기화로 인해 0의 값을 가지는 a5,a6이 각각 인가됨에 따라 상기 마스크 발생기 1105로부터의 M1,M2에 관계없이 상기 승산기 1130,1132의 출력은 상기 가산기 1140의 출력에 영향을 미치지 못한다.

즉, 상기 마스크 생성기 1105로부터 승산기 1130,1132로 입력되어지는 심볼열들이 어떤 값이든 상관없이 상기 승산기 1130,1132로부터는 0인 심볼열들이 출력된다. 따라서, 상기 승산기 1130,1132의 출력은 상기 배타적 가산기 1140에 입력되어도 아무런 영향을 미치지 않는다. 상기와 같이 a5,a6을 0으로 초기화하는 것은 상기 승산기 1130,1132의 출력을 차단하는 스위치 등작과 유사하다.

그러면, 상기 배타적 가산기 1140은 상기 승산기 1120, 1122, 1124, 1126, 1128, 1130, 1132로부터 출력되어진 길이 32인 모든 심볼 열들을 심볼 단위로 배타적 가산하여 천공기 1160으로 출력한다.

상기 배타적 가산기 1140으로부터 출력되는 길이 32의 부호화 심볼열(Ws)은 하기 < 수학식 17>로 표현될 수 있다.

수학식 17

$$Ws = (W1 \times 0) + (W2 \times 1) + (W4 \times 2) + (W8 \times 3) + (W16 \times 4)$$

이때, 상기 제어기 1110은 천공기 1160으로 (24,5) 부호에 대한 천공 위치를 천공하라는 제어신호를 출력한다. 그러면, 상기 천공기 1160은 천공위치가 저장되어 있는 메모리 1170으로부터 천공위치를 입력받아 상기 반복기 1150으로부터 제공되는 길이 32의 부호화 심볼들 중 8개의 해당 부호화 심볼들을 천공한다. 즉, 상기 천공기 1160은 상기 반복기 1150으로부터 제공되는 길이 32의 부호화 심볼들 중 상기 메모리 1170으로부터의 천공위치에 해당하는 8 심볼을 천공한다. 예컨대, 상기 메모리 1170으로부터의 천공위치가 0,4,8,12,16,20,24,28인 경우 상기 천공기 1160은 상기 길이 32의 부호화 심볼들 중 0,4,12,16,20,24,28번째 부호화 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

상기 (24,5) 부호기의 동작은 천공위치로 {0,4,8,12,16,20,24,28}을 가질 때에 대한 구성을 나타냈다. 하지만, 상기 (24,6) 부호기는 하드웨어의 복잡도를 줄이기 위해 {0,1,2,3,4,5,6,7}을 천공위치로서 가질 수 있다. 예컨대, 상기 메모리 1170으로부터의 천공위치가 0,1,2,3,4,5,6,7인 경우 상기 천공기 1160은 상기 길이 32의 부호화 심볼들 중 0,1,2,3,4,5,6,7번째 부호화 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

네 번째로, (24,4) 부호기로 동작할 때를 살펴보면, 4비트의 데이터를 지시자 a0,a1,a2,a3이 상기 부호기에 입력이 되고, 나머지 데이터를 지시자 a4,a5,a6은 초기화되어 입력된다. 상기 초기화에 의해 상기 a4,a5,a6으로는 0이 입력된다. 그러면, 상기 월시부호 발생기 1100은 길이 32인 월시부호 W1, W2, W4, W8을 출력한다. 상기 월시부호 발생기 1100으로부터 발생된 상기 W1, W2, W4, W8은 각각에 대응하는 승산기로 제공된다. 예컨대, 상기 W1은 승산기 1120으로 제공되며, 상기 W2는 승산기 1122로 제공된다. 상기 W4는 승산기 1124로 제공되며, 상기 W8은 승산기 1126으로 제공된다. 상기 월시부호 발생기 1100으로부터 발생되는 길이 32의 월시부호는 하기 < 표 33>과 같다.

[표 33]

월시번호	월시부호
W1	0101 0101 0101 0101 0101 0101 0101 0101
W2	0011 0011 0011 0011 0011 0011 0011 0011
W4	0000 1111 0000 1111 0000 1111 0000 1111
W8	0000 0000 1111 1111 0000 0000 1111 1111

한편, 상기 4개의 입력정보 비트들은 상기 승산기들 중 대응하는 각각의 상기 승산기로 제공된다. 상기 4개의 입력정보 비트들과 상기 승산기들의 대응 관계는 하기 < 표 34>에서 보이고 있는 바와 같다.

[표 34]

입력정보 비트	승산기
a0	승산기 1120
a1	승산기 1122
a2	승산기 1124
a3	승산기 1126

따라서, 상기 승산기들 각각으로 제공되는 입력정보 비트와 월시부호들은 하기 < 표 35> 과 같이 나타낼 수 있다.

[표 35]

승산기	월시부호	입력정보 비트
승산기 1120	W1 = 0101 0101 0101 0101 0101 0101 0101 0101	a0
승산기 1122	W2 = 0011 0011 0011 0011 0011 0011 0011 0011	a1
승산기 1124	W4 = 0000 1111 0000 1111 0000 1111 0000 1111	a2
승산기 1126	W8 = 0000 0000 1111 1111 0000 0000 1111 1111	a3

상기 승산기들은 상기 < 표 35> 에서 보이고 있는 바와 같이 자신에게로 제공되는 입력정보 비트를 월시부호를 구성하는 각각의 심볼들과 승산하여 가산기 1140으로 출력한다.

즉, 상기 승산기 1120은 a0을 W1 각각의 심볼마다 승산하여 가산기 1140으로 출력하고, 상기 승산기 1122는 a1을 W2 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력한다. 한편, 상기 승산기 1124는 a2를 W4 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력하며, 상기 승산기 1126은 a3을 W8 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력한다. 상기 가산기 1140은 배타적 가산기를 사용한다.

한편, 승산기 1128, 1130, 1132로는 상기 초기화로 인해 0의 값을 가지는 a4, a5, a6이 각각 인가됨에 따라 상기 월시부호 발생기 1100으로부터의 W16과 마스크 발생기 1105로부터의 M1, M2에 관계없이 상기 승산기 1128, 1130, 1132의 출력은 상기 가산기 1140의 출력에 영향을 미치지 못한다.

즉, 상기 월시부호 생성기 1100으로부터 승산기 1128로 입력되어지는 W16 심볼열들이 어떤 값이든 상관없이 상기 승산기 1128로부터는 0인 심볼열들이 출력된다. 또한, 상기 마스크 생성기 1105로부터 승산기 1130, 1132로 입력되어지는 심볼열들이 어떤 값이든 상관없이 상기 승산기 1130, 1132로부터는 0인 심볼열들이 출력된다. 따라서, 상기 승산기 1128, 1130, 1132의 출력은 상기 배타적 가산기 1140에 입력되어도 아무런 영향을 미치지 않는다. 상기와 같이 a4, a5, a6을 0으로 초기화하는 것은 상기 승산기 1128, 1130, 1132의 출력을 차단하는 스위치 동작과 유사하다.

그러면, 상기 배타적 가산기 1140은 상기 승산기 1120, 1122, 1124, 1126, 1128, 1130, 1132로부터 출력되어진 길이 32인 모든 심볼 열들을 심볼 단위로 배타적 가산하여 천공기 1160으로 출력한다.

상기 배타적 가산기 1140으로부터 출력되는 길이 32의 부호화 심볼열(Ws)은 하기 < 수학식 18> 로 표현될 수 있다.

수학식 18

$$Ws = (W1 \times 0) + (W2 \times 1) + (W4 \times 2) + (W8 \times 3)$$

이때, 상기 제어기 1110은 천공기 1160으로 (24,4) 부호에 대한 천공 위치를 천공하라는 제어신호를 출력한다. 그러면, 상기 천공기 1160은 천공위치가 저장되어 있는 메모리 1170으로부터 천공위치를 입력받아 상기 배타적 가산기 1140으로부터 제공되는 길이 32의 부호화 심볼들 중 8개의 해당 부호화 심볼들을 천공한다. 즉, 상기 천공기 1160은 상기 배타적 가산기 1140으로부터 제공되는 길이 32의 부호화 심볼들 중 상기 메모리 1170으로부터의 천공위치에 해당하

는 8 심볼을 천공한다. 예컨대, 상기 메모리 1170으로부터의 천공위치가 0,1,2,3,4,5,6,16인 경우 상기 천공기 1160은 상기 길이 32의 부호화 심볼들 중 0,1,2,3,4,5,6,16번째 부호화 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

다섯 번째로, (24,3) 부호기로 동작할 때를 살펴보면, 3비트의 데이터를 지시자 a0,a1,a2이 상기 부호기에 입력이 되고, 나머지 데이터를 지시자 a3,a4,a5,a6은 초기화되어 입력된다. 상기 초기화에 의해 상기 a3,a4,a5,a6으로는 0이 입력된다. 그러면, 상기 월시부호 발생기 1100은 길이 32인 월시부호 W1, W2, W4을 출력한다. 상기 월시부호 발생기 1100으로부터 발생된 상기 W1, W2, W4은 각각에 대응하는 승산기로 제공된다. 예컨대, 상기 W1은 승산기 1120으로 제공되며, 상기 W2는 승산기 1122로 제공된다. 상기 W4는 승산기 1124로 제공된다. 상기 월시부호 발생기 1100으로부터 발생되는 길이 32의 월시부호는 하기 < 표 36> 과 같다.

[표 36]

월시번호	월시부호
W1	0101 0101 0101 0101 0101 0101 0101 0101 0101
W2	0011 0011 0011 0011 0011 0011 0011 0011 0011
W4	0000 1111 0000 1111 0000 1111 0000 1111

한편, 상기 3개의 입력정보 비트들은 상기 승산기들 중 대응하는 각각의 상기 승산기로 제공된다. 상기 3개의 입력정보 비트들과 상기 승산기들의 대응 관계는 하기 < 표 37>에서 보이고 있는 바와 같다.

[표 37]

입력정보 비트	승산기
a0	승산기 1120
a1	승산기 1122
a2	승산기 1124

따라서, 상기 승산기들 각각으로 제공되는 입력정보 비트와 월시부호들은 하기 < 표 38> 와 같이 나타낼 수 있다.

[표 38]

승산기	월시부호	입력정보 비트
승산기 1120	W1 = 0101 0101 0101 0101 0101 0101 0101 0101	a0
승산기 1122	W2 = 0011 0011 0011 0011 0011 0011 0011 0011	a1
승산기 1124	W4 = 0000 1111 0000 1111 0000 1111 0000 1111	a2

상기 승산기들은 상기 < 표 38>에서 보이고 있는 바와 같이 자신에게로 제공되는 입력정보 비트를 월시부호를 구성하는 각각의 심볼들과 승산하여 가산기 1140으로 출력한다.

즉, 상기 승산기 1120은 a0을 W1 각각의 심볼마다 승산하여 가산기 1140으로 출력하고, 상기 승산기 1122는 a1을 W2 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력한다. 한편, 상기 승산기 1124는 a2를 W4 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력한다. 상기 가산기 1140은 배타적 가산기를 사용한다.

한편, 승산기 1126, 1128, 1130, 1132로는 상기 초기화로 인해 0의 값을 가지는 a3, a4, a5, a6이 각각 인가됨에 따라 상기 월시부호 발생기 1100으로부터의 W8, W16과 마스크 발생기 1105로부터의 M1, M2에 관계없이 상기 승산기 1126, 1128, 1130, 1132의 출력은 상기 가산기 1140의 출력에 영향을 미치지 못한다.

즉, 상기 월시부호 생성기 1100으로부터 승산기 1126로 입력되어지는 W8 심볼열들과 1128로 입력되어지는 W16 심볼열들이 어떤 값이든 상관없이 상기 승산기 1126, 1128로부터는 0인 심볼열들이 출력된다. 또한, 상기 마스크 생성기 1105로부터 승산기 1130, 1132로 입력되어지는 심볼열들이 어떤 값이든 상관없이 상기 승산기 1130, 1132로부터는 0인 심볼열들이 출력된다. 따라서, 상기 승산기 1126, 1128, 1130, 1132의 출력은 상기 배타적 가산기 1140에 입력되어도 아무런 영향을 미치지 않는다. 상기와 같이 a3,a4,a5,a6을 0으로 초기화하는 것은 상기 승산기 1126, 1128, 1130, 1132의 출력을 차단하는 스위치 동작과 유사하다.

그러면, 상기 배타적 가산기 1140은 상기 승산기 1120, 1122, 1124, 1126, 1128, 1130, 1132로부터 출력되어진 길이 32인 모든 심볼 열들을 심볼 단위로 배타적 가산하여 천공기 1160으로 출력한다.

상기 배타적 가산기 1140으로부터 출력되는 길이 32의 부호화 심볼열(Ws)은 하기 < 수학식 19>로 표현될 수 있다.

수학식 19

$$Ws = (W1 \times 0) + (W2 \times 1) + (W4 \times 2)$$

이때, 상기 제어기 1110은 천공기 1160으로 (24,3) 부호에 대한 천공 위치를 천공하라는 제어신호를 출력한다. 그러면, 상기 천공기 1160은 천공위치가 저장되어 있는 메모리 1170으로부터 천공위치를 입력받아 상기 배타적 가산기 1140으로부터 제공되는 길이 32의 부호화 심볼들 중 8개의 해당 부호화 심볼들을 천공한다. 즉, 상기 천공기 1160은 상기 배타적 가산기 1140으로부터 제공되는 길이 32의 부호화 심볼들 중 상기 메모리 1170으로부터의 천공위치에 해당하는 8 심볼을 천공한다. 예컨대, 상기 메모리 1170으로부터의 천공위치가 0,3,5,6,7,8,16,24인 경우 상기 천공기 1160은 상기 길이 32의 부호화 심볼들 중 0,3,5,6,7,8,16,24번째 부호화 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

여섯 번째로, (24,2) 부호기로 동작할 때를 살펴보면, 2비트의 데이터를 지시자 a0,a1이 상기 부호기에 입력이 되고, 나머지 데이터를 지시자 a2,a3,a4,a5,a6은 초기화되어 입력된다. 상기 초기화에 의해 상기 a2,a3,a4,a5,a6으로는 0이 입력된다. 그러면, 상기 월시부호 발생기 1100은 길이 32인 월시부호 W1, W2를 출력한다. 상기 월시부호 발생기 1100으로부터 발생된 상기 W1, W2는 각각에 대응하는 승산기로 제공된다. 예컨대, 상기 W1은 승산기 1120으로 제공되며, 상기 W2는 승산기 1122로 제공된다. 상기 월시부호 발생기 1100으로부터 발생되는 길이 32의 월시부호는 하기 < 표 39>과 같다.

[표 39]

월시번호	월시부호
W1	0101 0101 0101 0101 0101 0101 0101 0101
W2	0011 0011 0011 0011 0011 0011 0011 0011

한편, 상기 2개의 입력정보 비트들은 상기 승산기들 중 대응하는 각각의 상기 승산기로 제공된다. 상기 2개의 입력정보 비트들과 상기 승산기들의 대응 관계는 하기 < 표 40>에서 보이고 있는 바와 같다.

[표 40]

입력정보 비트	승산기
a0	승산기 1120
a1	승산기 1122

따라서, 상기 승산기들 각각으로 제공되는 입력정보 비트와 월시부호들은 하기 < 표 41> 와 같이 나타낼 수 있다.

[표 41]

승산기	월시부호	입력정보 비트
승산기 1120	W1 = 0101 0101 0101 0101 0101 0101 0101 0101	a0
승산기 1122	W2 = 0011 0011 0011 0011 0011 0011 0011 0011	a1

상기 승산기들은 상기 < 표 41> 에서 보이고 있는 바와 같이 자신에게로 제공되는 입력정보 비트를 월시부호를 구성하는 각각의 심볼들과 승산하여 가산기 1140으로 출력한다.

즉, 상기 승산기 1120은 a0을 W1 각각의 심볼마다 승산하여 가산기 1140으로 출력하고, 상기 승산기 1122는 a1을 W2 각각의 심볼마다 승산하여 상기 가산기 1140으로 출력한다. 상기 가산기 1140은 배타적 가산기를 사용한다.

한편, 승산기 1124, 1126, 1128, 1130, 1132로는 상기 초기화로 인해 0의 값을 가지는 a2, a3, a4, a5, a6이 각각 인가됨에 따라 상기 월시부호 발생기 1100으로부터의 W4, W8, W16과 마스크 발생기 1105로부터의 M1, M2에 관계 없이 상기 승산기 1124, 1126, 1128, 1130, 1132의 출력은 상기 가산기 1140의 출력에 영향을 미치지 못한다.

즉, 상기 월시부호 생성기 1100으로부터 승산기 1124로 입력되어지는 W4 심볼열들과 승산기 1126로 입력되어지는 W8 심볼열들과 1128로 입력되어지는 W16 심볼열들이 어떤 값이든 상관없이 상기 승산기 1124, 1126, 1128로부터는 0인 심볼열들이 출력된다. 또한, 상기 마스크 생성기 1105로부터 승산기 1130, 1132로 입력되어지는 심볼열들이 어떤 값이든 상관없이 상기 승산기 1130, 1132로부터는 0인 심볼열들이 출력된다. 따라서, 상기 승산기 1124, 1126, 1128, 1130, 1132의 출력은 상기 배타적 가산기 1140에 입력되어도 아무런 영향을 미치지 않는다. 상기와 같이 a2, a3, a4, a5, a6을 0으로 초기화하는 것은 상기 승산기 1124, 1126, 1128, 1130, 1132의 출력을 차단하는 스위치 등작과 유사하다.

그러면, 상기 배타적 가산기 1140은 상기 승산기 1120, 1122, 1124, 1126, 1128, 1130, 1132로부터 출력되어진 길이 32인 모든 심볼 열들을 심볼 단위로 배타적 가산하여 천공기 1160으로 출력한다.

상기 배타적 가산기 1140으로부터 출력되는 길이 32의 부호화 심볼열(Ws)은 하기 < 수학식 20> 로 표현될 수 있다.

수학식 20

$$Ws = (W1 \times 0) + (W2 \times 1)$$

이때, 상기 제어기 1110은 천공기 1160으로 (24,2) 부호에 대한 천공 위치를 천공하라는 제어신호를 출력한다. 그러면, 상기 천공기 1160은 천공위치가 저장되어 있는 메모리 1170으로부터 천공위치를 입력받아 상기 배타적 가산기 1140으로부터 제공되는 길이 32의 부호화 심볼들 중 8개의 해당 부호화 심볼들을 천공한다. 즉, 상기 천공기 1160은 상기 배타적 가산기 1140으로부터 제공되는 길이 32의 부호화 심볼들 중 상기 메모리 1170으로부터의 천공위치에 해당하는 8 심볼을 천공한다. 예컨대, 상기 메모리 1170으로부터의 천공위치가 0,4,8,12,16,20,24,28인 경우 상기 천공기 1160은 상기 길이 32의 부호화 심볼들 중 0,4,8,12,16,20,24,28번째 부호화 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

일곱 번째로, (24,1)부호기로 동작할 때를 살펴보면, 1비트의 데이터를 지시자 a0이 상기 부호기에 입력이 되고, 나머지 데이터를 지시자 a1,a2,a3,a4,a5,a6은 초기화되어 입력된다. 상기 초기화에 의해 상기 a1,a2,a3,a4,a5,a6으로는 0이 입력된다. 그러면, 상기 월시부호 발생기 1100은 길이 32인 월시부호 W1을 출력한다. 상기 월시부호 발생기 1100으로부터 발생된 상기 W1은 이에 대응하는 승산기로 제공된다. 예컨대, 상기 W1은 승산기 1120으로 제공된다. 상기 월시부호 발생기 1100으로부터 발생되는 길이 32인 월시부호는 하기 < 표 42> 과 같다.

[표 42]

월시번호	월시부호
W1	0101 0101 0101 0101 0101 0101 0101 0101 0101

한편, 상기 1개의 입력정보 비트들은 상기 승산기들 중 대응하는 각각의 상기 승산기로 제공된다. 상기 1개의 입력정보 비트들과 상기 승산기들의 대응 관계는 하기 < 표 43>에서 보이고 있는 바와 같다.

[표 43]

입력정보 비트	승산기
a0	승산기 1120

따라서, 상기 승산기들 각각으로 제공되는 입력정보 비트와 월시부호들은 하기 < 표 44>와 같이 나타낼 수 있다.

[표 44]

승산기	월시부호	입력정보 비트
승산기 1120	W1 = 0101 0101 0101 0101 0101 0101 0101 0101	a0

상기 승산기들은 상기 < 표 44>에서 보이고 있는 바와 같이 자신에게로 제공되는 입력정보 비트를 월시부호를 구성하는 각각의 심볼들과 승산하여 가산기 1140으로 출력한다.

즉, 상기 승산기 1120은 a0을 W1 각각의 심볼마다 승산하여 가산기 1140으로 출력한다. 상기 가산기 1140은 배타적 가산기를 사용한다.

한편, 승산기 1122, 1124, 1126, 1128, 1130, 1132로는 상기 초기화로 인해 0의 값을 가지는 a1, a2, a3, a4, a5, a6이 각각 인가됨에 따라 상기 월시부호 발생기 1100으로부터의 W2, W4, W8, W16과 마스크 발생기 1105로부터의 M1, M2에 관계없이 상기 승산기 1122, 1124, 1126, 1128, 1130, 1132의 출력은 상기 가산기 1140의 출력에 영향을 미치지 못한다.

즉, 상기 월시부호 생성기 1100으로부터 승산기 1122로 입력되어지는 W2 심볼열들과 승산기 1124로 입력되어지는 W4 심볼열들과 승산기 1126로 입력되어지는 W8 심볼열들과 1128로 입력되어지는 W16 심볼열들이 어떤 값이든 상관없이 상기 승산기 1122, 1124, 1126, 1128로부터는 0인 심볼열들이 출력된다. 또한, 상기 마스크 생성기 1105로부터 승산기 1130, 1132로 입력되어지는 심볼열들이 어떤 값이든 상관없이 상기 승산기 1130, 1132로부터는 0인 심볼열들이 출력된다. 따라서, 상기 승산기 1122, 1124, 1126, 1128, 1130, 1132의 출력은 상기 배타적 가산기 1140에 입력되어도 아무런 영향을 미치지 않는다. 상기와 같이 a1,a2,a3,a4,a5,a6을 0으로 초기화하는 것은 상기 승산기 1122, 1124, 1126, 1128, 1130, 1132의 출력을 차단하는 스위치 동작과 유사하다.

그러면, 상기 배타적 가산기 1140은 상기 승산기 1120, 1122, 1124, 1126, 1128, 1130, 1132로부터 출력되어진 길이 32인 모든 심볼 열들을 심볼 단위로 배타적 가산하여 천공기 1160으로 출력한다.

상기 배타적 가산기 1140으로부터 출력되는 길이 32의 부호화 심볼열(Ws)은 하기 < 수학식 21>로 표현될 수 있다.

수학식 21

$$Ws = (W1 \times 0)$$

이때, 상기 제어기 1110은 천공기 1160으로 (24,1) 부호에 대한 천공 위치를 천공하라는 제어신호를 출력한다. 그러면, 상기 천공기 1160은 천공위치가 저장되어 있는 메모리 1170으로부터 천공위치를 입력받아 상기 배타적 가산기 1140으로부터 제공되는 길이 32의 부호화 심볼들 중 8개의 해당 부호화 심볼들을 천공한다. 즉, 상기 천공기 1160은 상기 배타적 가산기 1140으로부터 제공되는 길이 32의 부호화 심볼들 중 상기 메모리 1170으로부터의 천공위치에 해당하는 8 심볼을 천공한다. 예컨대, 상기 메모리 1170으로부터의 천공위치가 1,3,5,7,9,11,13,15인 경우 상기 천공기 1160은 상기 길이 32의 부호화 심볼들 중 1,3,5,7,9,11,13,15번째 부호화 심볼들을 천공하여 24개의 부호화 심볼들을 출력하게 된다.

발명의 효과

전술한 바와 같이 본 발명은 확장된 일차 리드밀러 부호를 천공하는 장치 및 방법을 사용함으로서 최적의 성능과 최소의 복잡도를 가지는 데이터를 지시자 부호화 방법 및 장치를 구현할 수 있을 뿐만 아니라 최적의 부호어를 사용하는 효과를 가진다. 또한, 본 발명은 확장된 일차 리드밀러 부호를 천공하는 장치 및 방법을 사용함으로써 복호화 과정에서 역마다드를 사용할 수 있게 되어 하드웨어 복잡도(Hardware Complexity)를 최소화할 수 있으며, 오류 정정 성능에 있어서도 최적의 부호를 생성할 수 있는 효과가 있다. 마지막으로 본 발명은 (24,4) 부호화와 (24,7) 부호화를 겸용으로 지원함에 따라 보다 효율적인 부호화를 수행할 수 있는 장점이 있다.

(57) 청구의 범위

청구항 1.

6비트의 입력 정보비트들을 입력하고, 상기 6비트의 입력 정보비트들 각각을 길이 32의 서로 다른 다섯 개의 월시부호들과 하나의 마스크로 부호화하여 배타적 가산기에 의해 가산된 32개의 부호화 심볼들로 이루어진 부호화 심볼열을 24개의 부호화 심볼들로 이루어진 부호화 심볼열로 출력하는 이동통신시스템에서의 부호화 방법에 있어서,

상기 32개의 부호화 심볼들 중 0,4,8,12,16,20,24,28번째 심볼들을 천공하여 상기 24개의 부호화 심볼들로 이루어진 상기 부호화 심볼열을 출력하는 과정을 포함함을 특징으로 하는 상기 방법.

청구항 2.

제1항에 있어서,

상기 마스크는 0000 0000 1110 1000 1101 1000 1100 0000임을 특징으로 하는 상기 방법.

청구항 3.

6비트의 입력 정보비트들을 입력하고, 상기 6비트의 입력 정보비트들 각각을 길이 32의 서로 다른 다섯 개의 월시부호들과 하나의 마스크로 부호화하여 배타적 가산기에 의해 가산된 32개의 부호화 심볼들로 이루어진 부호화 심볼열을 24개의 부호화 심볼들로 이루어진 부호화 심볼열로 출력하는 이동통신시스템에서의 부호화 장치에 있어서,

상기 배타적 가산기로부터의 상기 32개의 부호화 심볼들 중 0,4,8,12,16,20,24,28번째 심볼들을 천공하여 상기 24개의 부호화 심볼들로 이루어진 상기 부호화 심볼열을 출력하는 천공기를 포함함을 특징으로 하는 상기 장치.

청구항 4.

제3항에 있어서,

상기 천공기에 의해 천공되는 천공 위치 0,4,8,12,16,20,24,28을 저장하는 메모리를 더 구비함을 특징으로 하는 상기 장치.

청구항 5.

제3항에 있어서,

상기 마스크는 0000 0000 1110 1000 1101 1000 1100 0000임을 특징으로 하는 상기 방법.

청구항 6.

6비트의 입력 정보비트들을 입력하고, 상기 6비트의 입력 정보비트들 각각을 길이 32의 서로 다른 다섯 개의 월시부호들과 하나의 마스크로 부호화하여 배타적 가산기에 의해 가산된 32개의 부호화 심볼들로 이루어진 부호화 심볼열을 2 4개의 부호화 심볼들로 이루어진 부호화 심볼열로 출력하는 이동통신시스템에서의 부호화 방법에 있어서,

상기 32개의 부호화 심볼들 중 0,1,2,3,4,5,6,7번째 심볼들을 천공하여 상기 24개의 부호화 심볼들로 이루어진 상기 부호화 심볼열을 출력하는 과정을 포함함을 특징으로 하는 상기 방법.

청구항 7.

제6항에 있어서,

상기 마스크는 0000 0000 1110 1000 1101 1000 1100 0000임을 특징으로 하는 상기 방법.

청구항 8.

6비트의 입력 정보비트들을 입력하고, 상기 6비트의 입력 정보비트들 각각을 길이 32의 서로 다른 다섯 개의 월시부호들과 하나의 마스크로 부호화하여 배타적 가산기에 의해 가산된 32개의 부호화 심볼들로 이루어진 부호화 심볼열을 2 4개의 부호화 심볼들로 이루어진 부호화 심볼열로 출력하는 이동통신시스템에서의 부호화 방법에 있어서,

상기 배타적 가산기로부터의 상기 32개의 부호화 심볼들 중 0,1,2,3,4,5,6,7번째 심볼들을 천공하여 상기 24개의 부호화 심볼들로 이루어진 상기 부호화 심볼열을 출력하는 천공기를 포함함을 특징으로 하는 상기 방법.

청구항 9.

제8항에 있어서,

상기 천공기에 의해 천공되는 천공 위치 0,1,2,3,4,5,6,7을 저장하는 메모리를 더 구비함을 특징으로 하는 상기 방법.

청구항 10.

제8항에 있어서,

상기 마스크는 0000 0000 1110 1000 1101 1000 1100 0000임을 특징으로 하는 상기 방법.

청구항 11.

1 내지 7비트의 입력 정보비트들을 입력하고, 상기 1 내지 상기 7비트의 입력 정보비트들을 24개의 부호화 심볼들로 이루어진 부호화 심볼열로 출력하는 이동통신시스템에서의 부호화 방법에 있어서,

상기 입력 정보비트들의 비트 수를 검사하는 과정과,

상기 입력 정보비트가 1비트로 검사될 시 상기 입력 정보비트를 상기 월시부호들 중 하나의 월시부호로 부호화하여 1회 반복하고, 상기 반복에 따른 64개의 부호화 심볼들로 이루어진 부호화 심볼열을 미리 결정된 소정 천공위치들의 심볼들을 천공하여 24개의 부호화 심볼들로 이루어진 부호화 심볼열로 출력하는 제1부호화 과정과,

상기 입력 정보비트들이 2 내지 7비트로 검사될 시 상기 입력 정보비트들 각각을 길이 32의 서로 다른 월시부호들과 마스크들로 부호화하여 배타적 가산기에 의해 가산된 32개의 부호화 심볼들로 이루어진 부호화 심볼열을 미리 결정된 소정 천공위치들의 심볼들을 천공하여 24개의 부호화 심볼들로 이루어진 부호화 심볼열로 출력하는 제2부호화 과정을 포함함을 특징으로 하는 상기 방법.

청구항 12.

제11항에 있어서,

상기 미리 결정된 소정 천공위치들은 상기 입력 정보비트들의 비트 수에 의해 선택함을 특징으로 하는 상기 방법.

청구항 13.

제12항에 있어서,

상기 입력 정보비트들의 비트 수에 의해 선택하는 상기 미리 결정된 소정 천공위치들은 하기 표 45와 같음을 특징으로 하는 상기 방법.

[표 45]

입력 정보비트 수	천공위치
1	짝수 번째 심볼들 및 1,3,5,7,9,11,13,15 번째 심볼
2	0,4,8,12,16,20,24,28 번째 심볼
3	0,3,5,6,7,8,16,24 번째 심볼
4	0,1,2,3,4,5,6,16 번째 심볼
5	0,1,2,3,4,5,6,7 번째 심볼
6	0,1,2,3,4,5,6,7 번째 심볼
7	0,1,2,3,4,5,6,7 번째 심볼

청구항 14.

제12항에 있어서,

상기 입력 정보비트들의 비트 수에 의해 선택하는 상기 미리 결정된 소정 천공위치들은 하기 표 45와 같음을 특징으로 하는 상기 방법.

[표 46]

입력 정보비트 수	천공 위치
1	짝수 번째 심볼들 및 1,3,5,7,9,11,13,15번째 심볼
2	0,4,8,12,16,20,24,28번째 심볼
3	0,3,5,6,7,8,16,24번째 심볼
4	0,1,2,3,4,5,6,16번째 심볼
5	0,4,8,12,16,20,24,28번째 심볼
6	0,4,8,12,16,20,24,28번째 심볼
7	0,4,8,12,16,20,24,28번째 심볼

청구항 15.

1 내지 7비트의 입력 정보비트들을 입력하고, 상기 1 내지 상기 7비트의 입력 정보비트들을 24개의 부호화 심볼들로 이루어진 부호화 심볼열로 출력하는 이동통신시스템에서의 부호화 방법에 있어서,

상기 1 비트 내지 7비트의 입력 정보비트들을 입력하고, 상기 입력 정보비트들 각각을 길이 32의 서로 다른 월시부호들과 마스크들로 부호화하여 배타적 가산기에 의해 가산된 32개의 부호화 심볼들을 출력하는 과정과,

상기 1 내지 7비트의 입력 정보비트들 각각에 대응하여 미리 결정된 소정 천공위치들 중 상기 입력된 입력 정보비트들의 비트 수에 의해 대응하는 하나의 천공위치를 선택하는 과정과,

상기 32개의 부호화 심볼들 중 상기 선택한 천공위치의 심볼들을 천공하여 24개의 부호화 심볼들로 출력하는 과정을 포함함을 특징으로 하는 상기 방법.

청구항 16.

제15항에 있어서,

상기 입력 정보비트들의 비트 수에 의해 선택하는 상기 미리 결정된 소정 천공위치들은 하기 표 47와 같음을 특징으로 하는 상기 방법.

[표 47]

입력 정보비트 수	천공 위치
1	1,3,5,7,9,11,13,15번째 심볼
2	0,4,8,12,16,20,24,28번째 심볼
3	0,3,5,6,7,8,16,24번째 심볼
4	0,1,2,3,4,5,6,16번째 심볼
5	0,1,2,3,4,5,6,7번째 심볼
6	0,1,2,3,4,5,6,7번째 심볼
7	0,1,2,3,4,5,6,7번째 심볼

청구항 17.

길이 32를 가지는 서로 다른 다섯 개의 월시부호들을 발생하는 월시부호 발생기와, 길이 32를 가지는 서로 다른 두 개의 마스크들을 발생하는 마스크 발생기를 구비하고, 1비트 내지 7비트의 입력 정보비트들을 부호화하여 24개의 부호화 심볼들을 출력하는 이동통신시스템에서의 부호화 장치에 있어서,

상기 입력 정보비트들로 입력되는 비트 수를 검사하여 상기 부호화를 위한 제어를 수행하는 제어부와,

상기 1비트 내지 7비트의 입력 정보비트들을 상기 월시부호 발생기와 상기 마스트 발생기로부터의 월시부호들 및 마스크들과 일대일로 승산하여 32개의 부호화 심볼들을 출력하는 승산기들과,

상기 승산기들로부터의 부호화 심볼열들을 배타적 가산하여 하나의 부호화 심볼열로 출력하는 배타적 가산기와,

상기 1비트의 입력 정보비트 입력에 따른 상기 제어부로부터의 제어에 의해 상기 배타적 가산기로부터의 상기 부호화 심볼들을 1회 반복하여 64개의 부호화 심볼들로 출력하고, 상기 2비트 내지 7비트의 입력 정보비트 입력에 따른 상기 제어부로부터의 제어에 의해 상기 배타적 가산기로부터의 상기 부호화 심볼들을 반복하지 않고 32개의 부호화 심볼들로 출력하는 반복기와,

상기 1 내지 7비트의 입력 정보비트들 각각에 대응하여 미리 결정된 소정 천공위치들을 저장하는 메모리와,

상기 제어부로부터의 제어에 의해 상기 메모리로부터 상기 입력 정보비트들로 입력되는 비트 수에 대응하는 천공위치를 독출하고, 상기 반복기로부터의 부호화 심볼들 중 상기 독출한 천공위치의 부호화 심볼들을 천공하여 상기 24개의 부호화 심볼들을 출력하는 천공기를 포함함을 특징으로 하는 상기 장치.

청구항 18.

제17항에 있어서,

상기 메모리에 상기 1 내지 7비트의 입력 정보비트들 각각에 대응하여 미리 결정된 소정 천공위치들은 하기 표 48와 같음을 특징으로 하는 상기 장치.

[표 48]

입력 정보비트 수	천공 위치
1	짝수 번째 심볼들 및 1,3,5,7,9,11,13,15번째 심볼
2	0,4,8,12,16,20,24,28번째 심볼
3	0,3,5,6,7,8,16,24번째 심볼
4	0,1,2,3,4,5,6,16번째 심볼
5	0,1,2,3,4,5,6,7번째 심볼
6	0,1,2,3,4,5,6,7번째 심볼
7	0,1,2,3,4,5,6,7번째 심볼

청구항 19.

제17항에 있어서,

상기 메모리에 상기 1 내지 7비트의 입력 정보비트들 각각에 대응하여 미리 결정된 소정 천공위치들은 하기 표 49와 같음을 특징으로 하는 상기 장치.

[표 49]

입력 정보비트 수	천공 위치
1	짝수 번째 심볼들 및 1,3,5,7,9,11,13,15번째 심볼
2	0,4,8,12,16,20,24,28번째 심볼
3	0,3,5,6,7,8,16,24번째 심볼
4	0,1,2,3,4,5,6,16번째 심볼
5	0,4,8,12,16,20,24,28번째 심볼
6	0,4,8,12,16,20,24,28번째 심볼
7	0,4,8,12,16,20,24,28번째 심볼

청구항 20.

길이 32를 가지는 서로 다른 다섯 개의 월시부호들을 발생하는 월시부호 발생기와, 길이 32를 가지는 서로 다른 두 개의 마스크들을 발생하는 마스크 발생기를 구비하고, 1비트 내지 7비트의 입력 정보비트들을 부호화하여 24개의 부호화 심볼들을 출력하는 이동통신시스템에서의 부호화 장치에 있어서,

상기 입력 정보비트들로 입력되는 비트 수를 검사하여 상기 부호화를 위한 제어를 수행하는 제어부와,

상기 1비트 내지 7비트의 입력 정보비트들을 상기 월시부호 발생기와 상기 마스트 발생기로부터의 월시부호들 및 마스크들과 일대일로 승산하여 32개의 부호화 심볼들을 출력하는 승산기들과,

상기 승산기들로부터의 부호화 심볼열들을 배타적 가산하여 하나의 부호화 심볼열로 출력하는 배타적 가산기와,

상기 1 내지 7비트의 입력 정보비트들 각각에 대응하여 미리 결정된 소정 천공위치들을 저장하는 메모리와,

상기 제어부로부터의 제어에 의해 상기 메모리로부터 상기 입력 정보비트들로 입력되는 비트 수에 대응하는 천공위치를 독출하고, 상기 반복기로부터의 부호화 심볼들 중 상기 독출한 천공위치의 부호화 심볼들을 천공하여 상기 24개의 부호화 심볼들을 출력하는 천공기를 포함함을 특징으로 하는 상기 장치.

청구항 21.

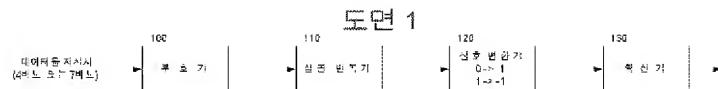
제20항에 있어서,

상기 메모리에 상기 1 내지 7비트의 입력 정보비트들 각각에 대응하여 미리 결정된 소정 천공위치들은 하기 표 50와 같음을 특징으로 하는 상기 장치.

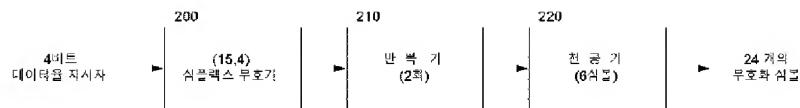
[표 50]

입력 정보비트 수	천공 위치
1	1,3,5,7,9,11,13,15번째 심볼
2	0,4,8,12,16,20,24,28번째 심볼
3	0,3,5,6,7,8,16,24번째 심볼
4	0,1,2,3,4,5,6,16번째 심볼
5	0,1,2,3,4,5,6,7번째 심볼
6	0,1,2,3,4,5,6,7번째 심볼
7	0,1,2,3,4,5,6,7번째 심볼

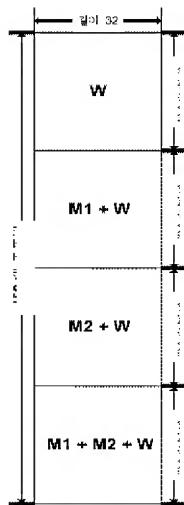
도면



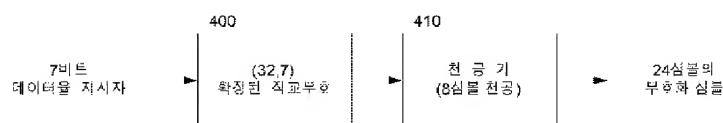
도면 2



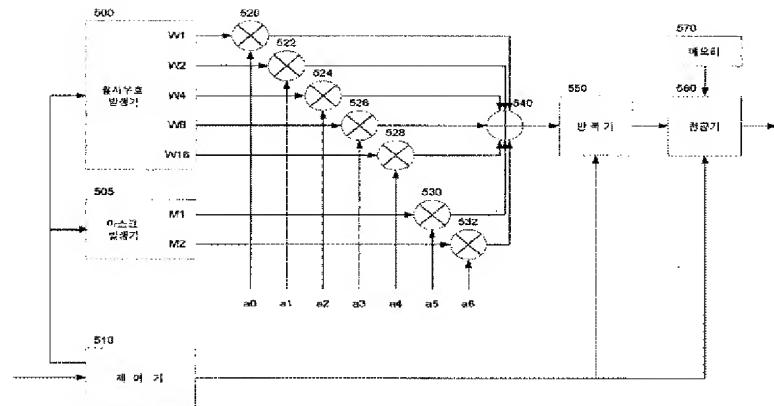
도면 3



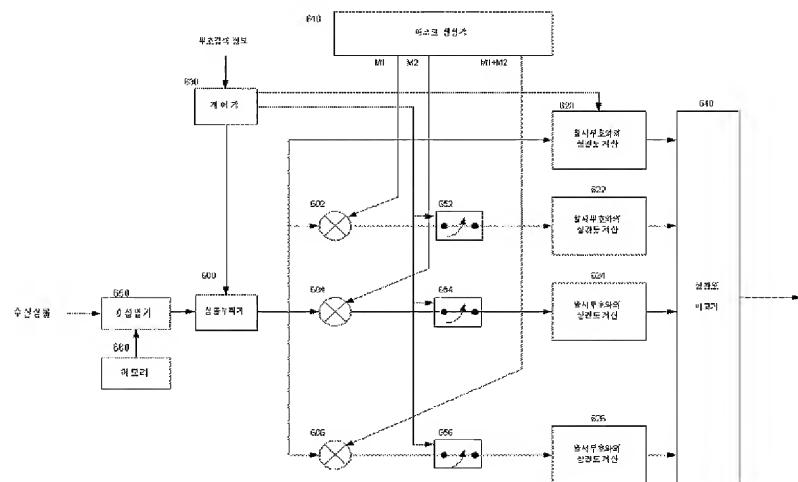
도면 4



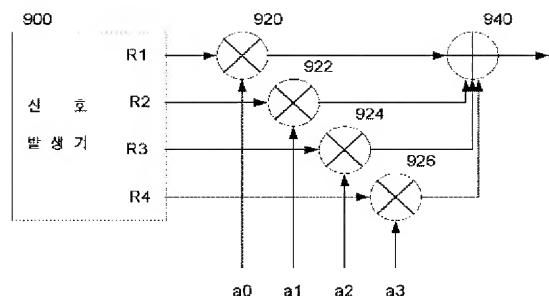
도면 5



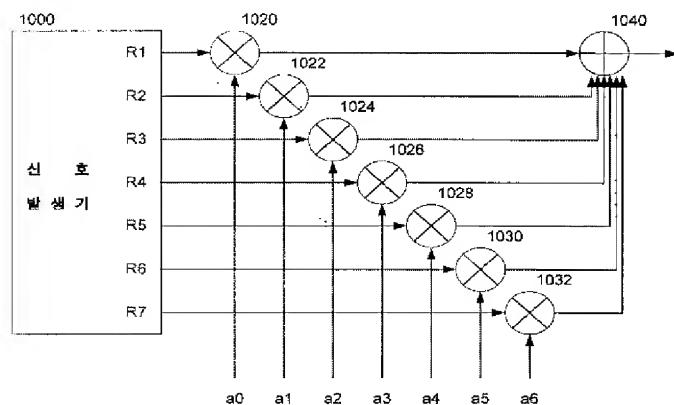
도면 6



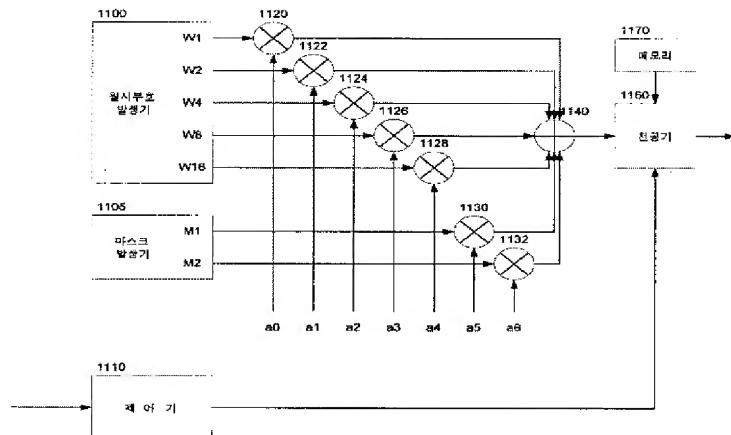
도면 7



도면 8



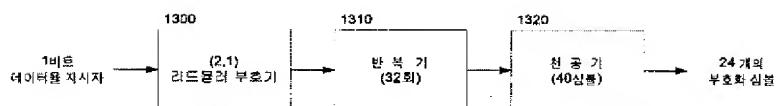
도면 9



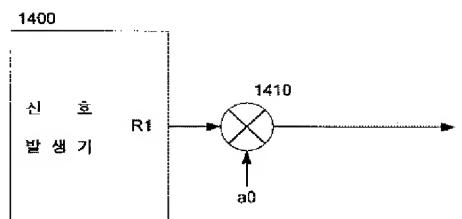
도면 10



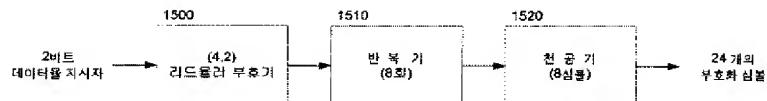
도면 11



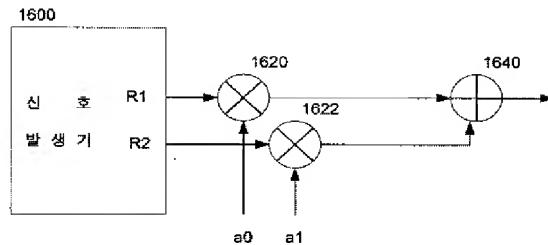
도면 12



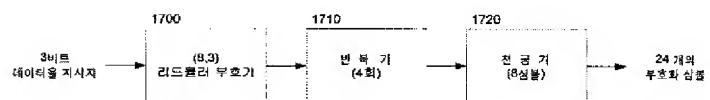
도면 13



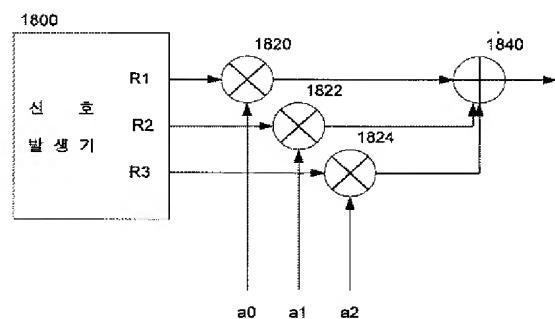
도면 14



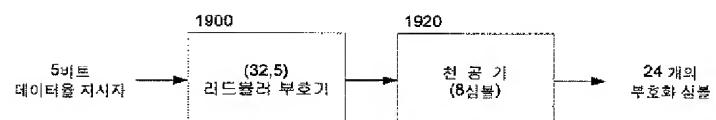
도면 15



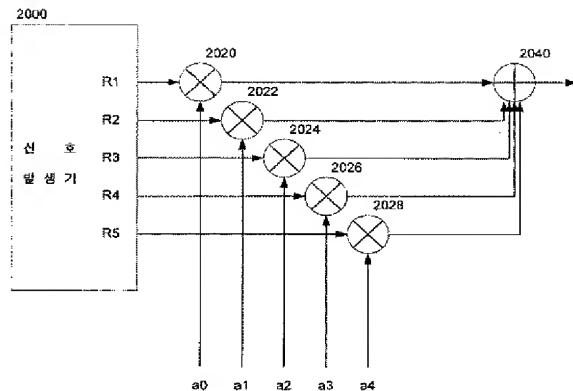
도면 16



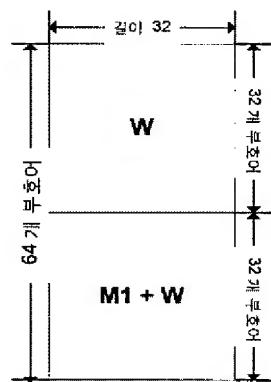
도면 17



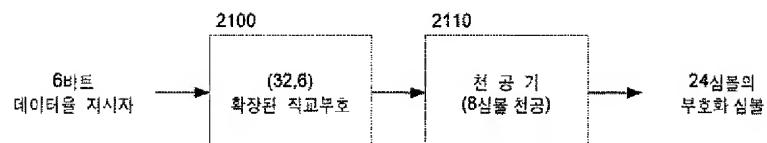
도면 18



도면 19



도면 20



도면 21

